



---

# B2B Commerce for Visualforce Developer Guide

Salesforce, Summer '24





# CONTENTS

<b>B2B COMMERCE FOR VISUALFORCE DEVELOPER GUIDE</b> .....	<b>1</b>
Custom Objects for B2B Commerce for Visualforce .....	<b>2</b>
Global API Classes for B2B Commerce for Visualforce .....	<b>282</b>
Data Service Provider Classes .....	<b>647</b>
Logic Service Provider Classes .....	<b>647</b>
REST API Endpoints for B2B Commerce for Visualforce .....	<b>1280</b>
Create Subscriber APIs for B2B Commerce for Visualforce .....	<b>1280</b>
Global Extension Point Classes for B2B Commerce for Visualforce .....	<b>1280</b>
Utility Classes for B2B Commerce for Visualforce .....	<b>1425</b>
Batch Processing Classes for B2B Commerce for Visualforce .....	<b>1476</b>
Trust Classes for B2B Commerce for Visualforce .....	<b>1485</b>
JavaScript Functions for B2B Commerce for Visualforce .....	<b>1493</b>



# B2B COMMERCE FOR VISUALFORCE DEVELOPER GUIDE

Each B2B Commerce for Visualforce implementation is unique, and solves use cases that are specific to each organization's requirements. If you can't fully develop your solution using default configuration options, you can extend different parts of the managed package's code architecture. Add fields to the managed package custom objects, override Apex API calls for retrieving and processing data, modify how the storefront front end renders, and more.

## IN THIS SECTION:

### [Custom Objects for B2B Commerce for Visualforce](#)

To support B2B Commerce for Visualforce use cases and avoid interfering with standard objects, the managed package installs several custom objects in your org. For example, B2B Commerce installs a custom product object, `ccrz__E_Product__c`, that exists in parallel with the standard product object. If your org already uses the standard product object, installing B2B Commerce doesn't affect how the standard product object is used. If you want, you can even synchronize your standard product object with the B2B Commerce custom product object.

### [Global API Classes for B2B Commerce for Visualforce](#)

B2B Commerce for Visualforce defines Apex API classes and methods that provide programmatic access to records of B2B Commerce custom objects. For example, the `ccrz.ccApiProduct` class provides methods for querying and returning data for your storefront's products.

### [Data Service Provider Classes](#)

B2B Commerce for Visualforce provides several default *data service providers*: Apex classes that query and transform specific data for global API methods. For example, the `ccrz.ccServiceCart` class defines how B2B Commerce queries and transforms cart data for a `ccrz.ccApiCart.fetch` method call. To modify how a default class behaves, you can write subscriber code that extends or overrides the default class.

### [Logic Service Provider Classes](#)

B2B Commerce for Visualforce provides several default *logic service providers*: Apex classes that describe the business logic for how specific global API methods behave. For example, the `ccrz.ccLogicCartAddTo` class defines how B2B Commerce adds products to a cart from a `ccrz.ccApiCart.addTo` method call. To modify the default logic for your organization, you can write subscriber code that extends or overrides the default logic.

### [REST API Endpoints for B2B Commerce for Visualforce](#)

B2B Commerce for Visualforce provides REST API endpoints for most of its global Apex API methods.

### [Create Subscriber APIs for B2B Commerce for Visualforce](#)

As part of your organization's customization and extension strategy, you can develop custom APIs that build on the default Apex classes and methods that B2B Commerce for Visualforce provides.

### [Global Extension Point Classes for B2B Commerce for Visualforce](#)

B2B Commerce for Visualforce provides several *global extension points*: Apex classes that are called from other code in the managed package and execute specific business logic for certain storefront functionality. For example, several extension point classes define behavior for loading data during checkout.

### [Utility Classes for B2B Commerce for Visualforce](#)

B2B Commerce for Visualforce provides several Apex classes with static fields, which effectively behave like global objects that expose certain storefront session data. Other classes and methods in the managed package can easily reference these static fields.

### [Batch Processing Classes for B2B Commerce for Visualforce](#)

B2B Commerce for Visualforce provides several Apex classes that provide global, asynchronous batch jobs. Each batch processing class extends the global `ccrz.ccBatchJob` base class.

### [Trust Classes for B2B Commerce for Visualforce](#)

B2B Commerce for Visualforce provides some default Apex classes for authoring unit tests and logging execution data.

### [JavaScript Functions for B2B Commerce for Visualforce](#)

B2B Commerce for Visualforce provides some proprietary JavaScript libraries or heavily modified third-party libraries that help render components and behavior on storefront pages. As part of your storefront's customization strategy, you can invoke, extend, or override these functions.

## Custom Objects for B2B Commerce for Visualforce

---

To support B2B Commerce for Visualforce use cases and avoid interfering with standard objects, the managed package installs several custom objects in your org. For example, B2B Commerce installs a custom product object, `ccrz__E_Product__c`, that exists in parallel with the standard product object. If your org already uses the standard product object, installing B2B Commerce doesn't affect how the standard product object is used. If you want, you can even synchronize your standard product object with the B2B Commerce custom product object.

To avoid name collisions with other custom objects, all B2B Commerce for Visualforce custom object and field names are prefixed with the `ccrz` namespace, as in `ccrz__E_Order__c`.

## Extend Custom Objects

You can extend B2B Commerce for Visualforce custom objects in the following ways:

- Add subscriber fields to any B2B Commerce custom object. Avoid duplicating the non-namespaced field name. For example, the `ccrz__E_Order__c` has the field `ccrz__Comments__c`. If you add a subscriber field to `ccrz__E_Order__c`, name it something other than `Comments__c`. You can add subscriber fields as part of a change set.
- Modify the default picklist values. You can only make these changes manually across all of your environments, and can't complete them as part of a change set.

## Field Data Types

When you're adding data to custom object records, each field on each object requires a specific data format. When you're returning existing data from those records with an API call, the value for each field comes back in a specific format. This guide identifies the input and output format for each field, using the following conventions:

- Input on object—The field's type for adding data to the object, such as Text or Checkbox. Some field types enforce a maximum length, such as 255 characters.
- Input or output with API—The data type for the field's value when passing data to or returning data from methods of the B2B Commerce for Visualforce API. For example, the value entered in a text-type field is stored in memory as a string primitive data type. For more details, see Primitive Data Types.

# Available Objects

## IN THIS SECTION:

### [Data Model for B2B Commerce for Visualforce](#)

The custom objects that B2B Commerce for Visualforce installs relate to each other in numerous ways. The buyer experience depends on establishing and maintaining these relationships.

### [Account \(B2B Commerce for Visualforce\)](#)

Represents a standard account object with custom fields that B2B Commerce for Visualforce adds.

### [Contact \(B2B Commerce for Visualforce\)](#)

Represents a standard contact object with custom fields that B2B Commerce for Visualforce adds.

### [User \(B2B Commerce for Visualforce\)](#)

Represents a standard user object with custom fields that B2B Commerce for Visualforce adds.

### [ccrz\\_\\_E\\_AccountAddressBook\\_\\_c](#)

Represents an address book entry created when a buyer adds an address from the Site Register or My Account pages. Each account address book is associated with a specific account and a specific contact address record.

### [ccrz\\_\\_E\\_AccountGroup\\_\\_c](#)

Represents an account group record, which associates one or more buyer accounts with entitled prices for products.

### [ccrz\\_\\_E\\_AccountGroupPriceList\\_\\_c](#)

Represents the assignment of a price list to an account group.

### [ccrz\\_\\_E\\_AccountSettings\\_\\_c](#)

Contains fields that support mixed pricing when you set up contract pricing on the storefront.

### [ccrz\\_\\_E\\_Attribute\\_\\_c](#)

Represents a parent or child attribute for products when the storefront uses attribute-driven commerce.

### [ccrz\\_\\_E\\_Cart\\_\\_c](#)

Represents a buyer's shopping cart, and specifies information about the cart's status.

### [ccrz\\_\\_E\\_CartItem\\_\\_c](#)

Represents an item in the buyer's shopping cart, including major and minor line items or applied coupons.

### [ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)

Represents a shipping group of cart items for line-level independence (LLI) checkout.

### [ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)

Represents the junction of a coupon and a shopping cart record.

### [ccrz\\_\\_E\\_Category\\_\\_c](#)

Represents a category record on the storefront.

### [ccrz\\_\\_E\\_CategoryI18N\\_\\_c](#)

Represents an internationalized category record for a specific locale.

### [ccrz\\_\\_E\\_CategoryMedia\\_\\_c](#)

Represents an image for a category that appears on the Product List page.

### [ccrz\\_\\_E\\_CategorySpec\\_\\_c](#)

Represents a spec value that's assigned to all products in a category.

[ccrz\\_\\_E\\_CompositeProduct\\_\\_c](#)

Represents composite and component product relationships for bundles, kits, dynamic kits, aggregated products, and assembly products.

[ccrz\\_\\_E\\_Configuration\\_\\_c](#)

Represents a storefront or global configuration setting.

[ccrz\\_\\_E\\_ConfigurationMeta\\_\\_c](#)

Represents metadata for an available storefront or global configuration setting.

[ccrz\\_\\_E\\_ConfigurationModule\\_\\_c](#)

Represents the definition of a module that contains global or storefront configuration settings.

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

Represents address details associated with carts, orders, invoices, subscriptions, payments, and other storefront data. A contact address is associated with an account address book, which references a specific account and specifies whether the address is for shipping or billing.

[ccrz\\_\\_E\\_Coupon\\_\\_c](#)

Represents a coupon record for the storefront.

[ccrz\\_\\_E\\_FeaturedProduct\\_\\_c](#)

Represents the details for a product shown in the Featured Products or Spotlight Products widget on your storefront's Home page.

[ccrz\\_\\_E\\_FeaturedProductI18N\\_\\_c](#)

Represents a featured product translated for a locale that your storefront supports.

[ccrz\\_\\_E\\_Invoice\\_\\_c](#)

Represents an invoice that B2B Commerce for Visualforce creates for a user after the user places an order.

[ccrz\\_\\_E\\_InvoiceItem\\_\\_c](#)

Represents a line item referenced from an invoice, such as a product or a fee.

[ccrz\\_\\_E\\_Menu\\_\\_c](#)

Represents a top-level entry in the storefront's navigation bar. Refresh the cached menu data whenever you modify menu records so that the storefront reflects the most recent menu data.

[ccrz\\_\\_E\\_MenuI18N\\_\\_c](#)

Represents an internationalized menu record for a specific locale.

[ccrz\\_\\_E\\_MenuItem\\_\\_c](#)

Represents a nested menu entry in a menu in the storefront's navigation bar. Refresh the cached menu data whenever you modify menu item records so that the storefront reflects the most recent menu data.

[ccrz\\_\\_E\\_MenuItemI18N\\_\\_c](#)

Represents an internationalized menu item record for a specific locale.

[ccrz\\_\\_E\\_Order\\_\\_c](#)

Represents a buyer's completed order, based on a shopping cart that the buyer checked out with.

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

Represents an item in a buyer's completed order, including major and minor line items or applied coupons.

[ccrz\\_\\_E\\_OrderItemGroup\\_\\_c](#)

Represents a shipping group of order items for line-level independence (LLI) checkout.

[ccrz\\_\\_E\\_OrderShipment\\_\\_c](#)

Represents shipment details for an order.



[ccrz\\_\\_E\\_OrderShipmentItem\\_\\_c](#)

Represents shipment details for an item in a buyer's completed order.

[ccrz\\_\\_E\\_OrderTerm\\_\\_c](#)

Represents the terms and conditions that a buyer acknowledges and accepts during checkout.

[ccrz\\_\\_E\\_PageLabel\\_\\_c](#)

Represents a page label record, which defines buyer-facing text on the storefront, such as headings, buttons, and error messages. Each page label record is specific to one storefront and one locale.

[ccrz\\_\\_E\\_PageLabel18N\\_\\_c](#)

Represents a page label translated for a locale that your storefront supports.

[ccrz\\_\\_E\\_PageSection\\_\\_c](#)

Represents a page section record, which specifies extra product content—product specifications, related products, documentation, and so on—to show on the Product Detail page.

[ccrz\\_\\_E\\_PriceGroup\\_\\_c](#)

Determines alternative pricing for accounts outside of account groups and price lists. For example, use a price group for contract pricing, special pricing for industry segments, or other context-specific pricing.

[ccrz\\_\\_E\\_PriceGroupAccount\\_\\_c](#)

Represents the assignment of a price group to an account.

[ccrz\\_\\_E\\_PriceGroupItem\\_\\_c](#)

Represents a product included in a price group.

[ccrz\\_\\_E\\_PriceGroupPricelist\\_\\_c](#)

Represents the assignment of a price list to a price group.

[ccrz\\_\\_E\\_Pricelist\\_\\_c](#)

Represents a price list record, which assigns prices to products for a particular account group. Each price list can apply to only one currency.

[ccrz\\_\\_E\\_PricelistItem\\_\\_c](#)

Represents a specific price for a product in a price list.

[ccrz\\_\\_E\\_Product\\_\\_c](#)

Represents a product record for the storefront.

[ccrz\\_\\_E\\_ProductCategory\\_\\_c](#)

Represents the assignment of a product to a category.

[ccrz\\_\\_E\\_ProductGuide\\_\\_c](#)

Represents a product guide used for a dynamic kit.

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

Represents a flat data structure for one product assigned to one category and one locale. A product index record includes the product's pricing information and searchable data, such as SKU, short and long description, and SEO keywords. B2B Commerce for Visualforce references product index records for quickly populating content on the Product List page version 2.0.

[ccrz\\_\\_E\\_ProductInventoryItem\\_\\_c](#)

Represents the available inventory quantity for a product.

[ccrz\\_\\_E\\_ProductItem18N\\_\\_c](#)

Represents an internationalized product record for a specific locale.

[ccrz\\_\\_E\\_ProductMedia\\_\\_c](#)

Represents an image or documentation file for a product.

[ccrz\\_\\_E\\_ProductSpec\\_\\_c](#)

Represents the value of a product's specification, such as Stainless Steel for a spec named Material.

[ccrz\\_\\_E\\_ProductSpecIndex\\_\\_c](#)

Represents an instance of a single spec value for a particular product, cached for quickly retrieving spec values on the storefront.

[ccrz\\_\\_E\\_ProductTab\\_\\_c](#)

Represents custom rich text content to display as a page section on the Product Detail page.

[ccrz\\_\\_E\\_Promo\\_\\_c](#)

Represents a promotion record for a storefront.

[ccrz\\_\\_E\\_Promol18n\\_\\_c](#)

Represents a localized promotion record.

[ccrz\\_\\_E\\_PromotionAccountGroupFilter\\_\\_c](#)

Represents an account group that a promotion record includes or excludes, which targets the promotion at a specific set of users.

[ccrz\\_\\_E\\_PublicCache\\_\\_c](#)

Represents cached data for content that's publicly available and specific to a storefront, page, and locale.

[ccrz\\_\\_E\\_PublicCacheContent\\_\\_c](#)

Represents the content of a parent public cache record.

[ccrz\\_\\_E\\_RelatedProduct\\_\\_c](#)

Represents a relationship between two products, such as specifying that one product in your catalog complements or replaces another product.

[ccrz\\_\\_E\\_RelatedProductGroup\\_\\_c](#)

This object is available in the managed package, but isn't used for any default functionality.

[ccrz\\_\\_E\\_Seller\\_\\_c](#)

Represents a seller hosted on your storefront whose products are available for search and purchase.

[ccrz\\_\\_E\\_Sellerl18N\\_\\_c](#)

Represents a localized seller record.

[ccrz\\_\\_E\\_ShippingRate\\_\\_c](#)

Represents a shipping cost calculation for a product, based on the product's specified shipping weight, a shipping carrier, and a shipping method.

[ccrz\\_\\_E\\_Spec\\_\\_c](#)

Represents the definition for a specification, which identifies a type of physical, operational, or performance measurement for a product. The [ccrz\\_\\_E\\_ProductSpec\\_\\_c](#) object defines a specification value for a particular product.

[ccrz\\_\\_E\\_SiteIndex\\_\\_c](#)

Represents the hashed definition of a single category's friendly URL and its associated breadcrumb path for a specific storefront and locale.

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

Represents payment details saved in the My Wallet section of the My Account page.

[ccrz\\_\\_E\\_StorefrontAssociation\\_\\_c](#)

Represents a link between a particular account, a particular storefront, and a particular Experience Cloud site.

[ccrz\\_\\_E\\_SubProdTerm\\_\\_c](#)

Represents subscription options that you make available for a specific product.

[ccrz\\_\\_E\\_SubProdTermI18N\\_\\_c](#)

Represents subscription options that you make available for a specific product, localized for a specific storefront locale.

[ccrz\\_\\_E\\_Subscriber\\_Page\\_\\_c](#)

Represents a custom page that you add to the storefront.

[ccrz\\_\\_E\\_Subscription\\_\\_c](#)

Represents a buyer's selected subscription after they place an order.

[ccrz\\_\\_E\\_TaxRate\\_\\_c](#)

Represents a tax percentage for a state in a particular country or territory. The Checkout page applies this tax rate to all taxable products, depending on the specified shipping address.

[ccrz\\_\\_E\\_Term\\_\\_c](#)

Represents terms and conditions shown to the buyer during checkout.

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

Represents a payment made when a buyer completes checkout.

## Data Model for B2B Commerce for Visualforce

The custom objects that B2B Commerce for Visualforce installs relate to each other in numerous ways. The buyer experience depends on establishing and maintaining these relationships.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [Data Model](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## Account (B2B Commerce for Visualforce)

Represents a standard account object with custom fields that B2B Commerce for Visualforce adds.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiAccount`

**Fetch sizing:** To specify which fields the `ccrz.ccApiAccount.fetch` method returns for each account record, [request a specific data size](#) for the `ccrz.ccApiAccount.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAccount.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields Added by B2B Commerce for Visualforce

### Account Group

The account group that the account belongs to.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AccountGroup__c</code>	Lookup Relationship ( <a href="#">ccrz__E_AccountGroup__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	<code>EAccountGroupR</code>	ID

### Custom Field 1

A custom field that you can implement in your solution.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CustomField1__c</code>	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	<code>customField1</code>	String

### Custom Field 2

A custom field that you can implement in your solution.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CustomField2__c</code>	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	<code>customField2</code>	String

### Data ID

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	ccrz__DataId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

#### Prevent Address Book Saves

Whether the storefront user associated with the account can add, edit, or delete addresses in the Manage Address Book section of the My Account page. For usage, see [Allow a Buyer to Manage Shipping and Billing Addresses on the My Account Page](#).

Usage	Field Name	Data Type
Input on object	ccrz__PreventAddrBookSave__c	Checkbox
Input or output with API (with <b>SZ_S</b> or larger)	preventAddrBookSave	Boolean

#### Prevent New Addresses at Checkout

Whether a storefront user associated with the account can enter a new shipping address in the User Information section of the Checkout page. For usage, see [Specify Whether a Buyer Can Modify Addresses on the Checkout Page](#).

Usage	Field Name	Data Type
Input on object	ccrz__PreventNewAddrCheckout__c	Checkbox
Input or output with API (with <b>SZ_S</b> or larger)	preventNewAddrCheckout	Boolean

#### Prevent New Checkout Billing Address

Whether a storefront user associated with the account can enter a new billing address in the User Information section of the Checkout page. For usage, see [Specify Whether a Buyer Can Modify Addresses on the Checkout Page](#).

Usage	Field Name	Data Type
Input on object	ccrz__PreventNewBillAddrCO__c	Checkbox
Input or output with API (with <b>SZ_S</b> or larger)	preventNewBillAddrCO	Boolean

#### TaxExemptAccount

Whether the account is exempt from tax calculation during checkout. For more information, see [ccrz.cc\\_hk\\_TaxCalculation.skipTaxCalculation](#).

Usage	Field Name	Data Type
Input on object	ccrz__TaxExemptAccount__c	Checkbox

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_S</b> or larger)	taxExemptAccount	Boolean

#### Used by CloudCraze Customer Portal

Used in account sharing rules to identify portal user accounts.

Usage	Field Name	Data Type
Input on object	ccrz__UsedByCloudcraze__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Source System	ccrz__SourceSystem__c	Text (100)

SEE ALSO:

[ccrz\\_\\_E\\_AccountGroup\\_\\_c](#)

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

[Salesforce Help Enable Buyer Entitlements with Account Groups](#)

[Salesforce Help Show Account, Contact, and User Information on the My Account Page](#)

## Contact (B2B Commerce for Visualforce)

Represents a standard contact object with custom fields that B2B Commerce for Visualforce adds.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiContact`

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

The API doesn't return data for any of the custom fields that B2B Commerce for Visualforce adds.

## Fields Added by B2B Commerce for Visualforce

### Company Name

Included in the default Request for Quote form.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CompanyName__c</code>	Text (80)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Data ID

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__DataId__c</code>	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Level

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Level__c</code>	Picklist
Input or output with API (with any data size)	The API doesn't return data for this field.	

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Company Type	ccrz__CompanyType__c	Picklist
Portal User	ccrz__IsPortalUser__c	Checkbox
Languages	ccrz__Languages__c	Text (100)
Self Registered	ccrz__SelfRegistered__c	Checkbox

SEE ALSO:

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

[Salesforce Help Show Account, Contact, and User Information on the My Account Page](#)

## User (B2B Commerce for Visualforce)

Represents a standard user object with custom fields that B2B Commerce for Visualforce adds.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** `ccrz.ccApiUser`

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

### Fields Added by B2B Commerce for Visualforce

#### **Company Type**

Options for the [Company Type picklist on the Site Register page](#).



Usage	Field Name	Data Type
Input on object	ccrz__CompanyType__c	Picklist
Input or output with API (with any data size)	companyType	String

### Contact Type Role

Optional role of the user in your organization.

Usage	Field Name	Data Type
Input on object	ccrz__ContactTypeRole__c	Picklist
Input or output with API (with any data size)	contactTypeRole	String

### Currency

User's currency. This field is used in orgs that don't have multiple currencies enabled.



**Tip:** This field provides the value for [the user's currency shown on the My Account page](#). If your solution integrates storefront users from another system, or if you create storefront users manually, make sure to supply a value for this field. Leaving this field blank can result in unexpected behavior when viewing currency information on the My Account page.

Usage	Field Name	Data Type
Input on object	ccrz__CurrencyCode__c	Text (30)
Input or output with API (with any data size)	ccCurrencyCode	String

### Data ID

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	ccrz__DataId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Tax Exempt

Whether the user is exempt from tax calculations during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__Tax_Exempt__c	Checkbox

Usage	Field Name	Data Type
Input or output with API (with any data size)	taxExempt	Boolean

**User Industry**

Optional industry for the user.

Usage	Field Name	Data Type
Input on object	ccrz__User_Industry__c	Picklist
Input or output with API (with any data size)	userIndustry	String

**User Salutation**

User's salutation, which is specified from the [Title field of the Site Register page](#).

Usage	Field Name	Data Type
Input on object	ccrz__User_Salutation__c	Picklist
Input or output with API (with any data size)	userSalutation	String

## SEE ALSO:

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

[Salesforce Help Show Account, Contact, and User Information on the My Account Page](#)

**ccrz\_\_E\_AccountAddressBook\_\_c**

Represents an address book entry created when a buyer adds an address from the Site Register or My Account pages. Each account address book is associated with a specific account and a specific contact address record.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiAddressBook](#)

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

## Fields

### AccountID

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__AccountId__c	Text (50)
Input or output with API (with any data size)	accountId	String

### Address Book ID

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__AddressBookId__c	Text (80)
Input or output with API (with any data size)	addressBookId	String

### Address Type

Whether the associated contact address is for billing or shipping.

Usage	Field Name	Data Type
Input on object	ccrz__AddressType__c	Picklist
Input or output with API (with any data size)	addressType	String

### CC Contact Address

The specific address details associated with this address book entry.

Usage	Field Name	Data Type
Input on object	ccrz__E_ContactAddress__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with any data size)	EContactAddress	ID

**Custom Object Label Name**

Unique record ID for the account address book.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfdcName	String

**Default**

Whether the address is selected by default in billing or shipping address picklists on the storefront, depending on the record's Address Type.



**Note:** B2B Commerce for Visualforce doesn't enforce a limit of only one default billing or shipping address. You can enable Default on multiple account address book records. When multiple default billing or shipping addresses exist, B2B Commerce references whichever record was most recently edited to use as the default.

Usage	Field Name	Data Type
Input on object	ccrz__Default__c	Checkbox
Input or output with API (with any data size)	default	Boolean

**Owner**

ID of the user that owns the account address book record.

Usage	Field Name	Data Type
Input on object	ccrz__Owner__c	Lookup Relationship ( <a href="#">User</a> )
Input or output with API (with any data size)	owner	ID

**TypeReadOnly**

ID of the user that owns the account address book record.

Usage	Field Name	Data Type
Input on object	ccrz__TypeReadOnly__c	Checkbox
Input or output with API (with any data size)	typeReadOnly	Boolean

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Account	ccrz__Account__c	Lookup Relationship ( <a href="#">Account</a> )

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[Salesforce Help Account Address Books and Contact Addresses for B2B Commerce for Visualforce Buyers](#)

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

## ccrz\_\_E\_AccountGroup\_\_c

Represents an account group record, which associates one or more buyer accounts with entitled prices for products.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** The API doesn't directly query records of this object.

Instead, the `ccrz.ccApiAccount` return data includes the `EAccountGroupR` related list when the `ccrz.ccApiAccount.ENTITYNAME` input map key [requests a medium or larger data size](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAccount.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

**Field sizing:** Each record returned in the related list always returns the same fields, regardless of requested data size.

### Fields

**Account Group Name**

Name of the account group.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfdcName	String

**Account Group ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__AccountGroupID__c	Text (128)
Input or output with API (with any data size)	accountGroupID	String

**Data ID**

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	ccrz__DataId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Desc**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__Desc__c	Text (255)
Input or output with API (with any data size)	desc	String

**PriceList Selection Method**

How B2B Commerce for Visualforce selects the price list that's used for determining the entitled products and prices for buyers in the account group. Selecting a price list is necessary only when the account group has multiple price lists assigned. For picklist values and usage, see [Price List Selection Methods for an Account Group](#).

Usage	Field Name	Data Type
Input on object	ccrz__PriceListSelectionMethod__c	Picklist
Input or output with API (with any data size)	priceListSelectionMethod	String

**Theme Name**

The name of a unique storefront theme to show only for buyers in the account group. This functionality is implemented by the [ccrz.cc\\_hk\\_DynamicTheme extension point class](#). For more information, see:

- [Show a Unique Storefront Theme for an Account Group](#)
- [Style Your Storefront with a Theme](#)

Usage	Field Name	Data Type
Input on object	ccrz__ThemeName__c	Text (255)
Input or output with API (with any data size)	themeName	String

## SEE ALSO:

- [Account \(B2B Commerce for Visualforce\)](#)
- [Salesforce Help Enable Buyer Entitlements with Account Groups](#)
- [Salesforce Help Price List Selection Methods for an Account Group](#)
- [Salesforce Help Show a Unique Storefront Theme for an Account Group](#)
- [Salesforce Help Style Your Storefront with a Theme](#)

## ccrz\_\_E\_AccountGroupPricelist\_\_c

Represents the assignment of a price list to an account group.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** The API doesn't directly query records of this object.

### Fields

**Account Group**

The account group assigned to the price list.

Usage	Field Name	Data Type
Input on object	ccrz__AccountGroup__c	Lookup Relationship ( <a href="#">ccrz__E_AccountGroup__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Account Group Pricelist ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__AccountGroupPricelistID__c	Text (128)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Acct Group Pricelist Number**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Data ID**

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccrzApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	ccrz__DataId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Enable**

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enable__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	



**End Date**

Date when the relationship between account group and price list is no longer valid.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Pricelist**

The price list that the account group's assigned to.

Usage	Field Name	Data Type
Input on object	ccrz__Pricelist__c	Lookup Relationship ( <a href="#">ccrz__E_Pricelist__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Sequence**

Priority of the price list for the account group. This field is used only when the account group's price list selection method is Sequence Order. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_price\\_list\\_selection\\_method.htm](https://help.salesforce.com/articleView?id=b2b_commerce_price_list_selection_method.htm).

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Start Date**

Date when the relationship between account group and price list is valid.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with any data size)	The API doesn't return data for this field.	

## SEE ALSO:

[Salesforce Help Price List Selection Methods for an Account Group](#)

## ccrz\_\_E\_AccountSettings\_\_c

Contains fields that support mixed pricing when you set up contract pricing on the storefront.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** The API doesn't directly query records of this object.

### Fields

#### Account

The account that these settings apply to.

Usage	Field Name	Data Type
Input on object	ccrz__Account__c	Master-Detail Relationship ( <a href="#">Account</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

#### Allow Mixed Pricing Carts

[Honor Custom Prices Negotiated in a Contract](#)

Usage	Field Name	Data Type
Input on object	ccrz__AllowMixedPricingCarts__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

#### CC Account Setting Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

SEE ALSO:

[Salesforce Help Honor Custom Prices Negotiated in a Contract](#)

## ccrz\_\_E\_Attribute\_\_c

Represents a parent or child attribute for products when the storefront uses attribute-driven commerce.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** `ccrz.ccApiAttribute`

**Fetch sizing:** To specify which fields the `ccrz.ccApiAttribute.fetch` method returns for each attribute record, [request a specific data size](#) for the `ccrz.ccApiAttribute.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAttribute.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

### Fields

#### Attribute ID

A unique ID that associates the record with an external system.

 **Note:** This field is required. B2B Commerce for Visualforce uses this ID as a key in JSON-formatted attribute pricing data.

Usage	Field Name	Data Type
Input on object	ccrz__AttributeID__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	attributeID	String

**CC Attribute Name**

Attribute record name.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Display Name**

The attribute name displayed on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__DisplayName__c	Text Area (Rich) (1024)
Input or output with API (with <b>SZ_M</b> or larger)	displayName	String

**Effective End**

Date when this attribute is no longer available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveEnd__c	Date/Time
Input or output with API (with <b>SZ_XL</b> )	effectiveEnd	Datetime

**Effective Start**

Date when this attribute is available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveStart__c	Date/Time
Input or output with API (with <b>SZ_XL</b> )	effectiveStart	Datetime

**Parent Attribute**

Date when this attribute is available.

Usage	Field Name	Data Type
Input on object	ccrz__ParentAttribute__c	Lookup Relationship ( ccrz__E_Attribute__c)
Input or output with API (with <b>SZ_S</b> or larger)	parentAttribute	ID
	parentAttributeR	Map<String, Object> that describes the parent attribute.

### Sequence

The order that a child attribute displays under a parent attribute. Lower Sequence values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with <b>SZ_L</b> or larger)	sequence	Decimal

### SKU Fragment

The unique SKU for a child attribute. The selected child attribute SKUs form the product's SKU.

Usage	Field Name	Data Type
Input on object	ccrz__SKUFragment__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	SKUFragment	String

SEE ALSO:

[Salesforce Help Set Up Attribute-Based Products](#)

## ccrz\_\_E\_Cart\_\_c

Represents a buyer's shopping cart, and specifies information about the cart's status.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

Class used with: [ccrz.ccApiCart](#)

**Fetch sizing:** To specify which fields the [ccrz.ccApiCart.fetch](#) method returns for each cart record, [request a specific data size](#) for the `ccrz.ccApiCart.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Account

The account associated with the user assigned in the `User` field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Account__c</code>	Lookup Relationship ( <a href="#">Account</a> )
Input or output with API (with <b>SZ_XL</b> )	<code>account</code>	ID

### Active Cart

Whether the cart is *active* for the buyer's storefront session. Only one cart can be active per account at one time. For more information about active and inactive carts, see [Considerations for Supporting a Buyer's Cart Experience](#).

Usage	Field Name	Data Type
Input on object	<code>ccrz__ActiveCart__c</code>	Checkbox
Input or output with API (with <b>SZ_S</b> or larger)	<code>activeCart</code>	Boolean

### Adjustment Amount

Amount to adjust the order price by after the order is placed. This amount is specified on the cart, and passed to the corresponding [ccrz\\_\\_E\\_Order\\_\\_c](#) record. You can use your own back-end processing to apply this adjustment without changing the values of other fields.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AdjustmentAmount__c</code>	Currency (16, 2)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_L</b> or larger)	adjustmentAmount	Decimal

**Adjustment Reason**

Text that explains or justifies an adjustment amount. This text is specified on the cart, and passed to the corresponding [ccrz\\_\\_E\\_Order\\_\\_c](#) record.

Usage	Field Name	Data Type
Input on object	ccrz__AdjustmentReason__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	adjustmentReason	String

**Anonymous ID**

Whether the user associated with the cart is a guest or registered buyer. If a buyer creates a cart as a guest and then logs in, this field is evaluated when transferring the guest cart to the authenticated buyer.

Usage	Field Name	Data Type
Input on object	ccrz__AnonymousId__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	anonymousID	Boolean

**Asynchronous Job ID**

The ID of the asynchronous Apex job that's executing for the cart.

 **Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	ccrz__AsynchronousJobID__c	Text (255)
Input or output with API (with <b>SZ_M</b> ) or larger	asynchronousJobID	String

**Bill To**

The billing address associated with the cart. This record is a unique copy of the billing address specified during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__BillTo__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with <b>SZ_XL</b> )	billTo	ID

**Cart Auto Number**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	ID

**Cart ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__CartId__c	Text (128)
Input or output with API (with <b>SZ_XL</b> )	cartId	String

**Cart Status**

Whether the cart is open:

- Open
- Closed after a corresponding order is placed
- Abandoned
- Processing, which indicates that the `ccrz.ccApiCart.placeAsync` method is placing the order using asynchronous Apex jobs.
- Repricing, which indicates that pricing logic is reevaluating the cart's items and prices.



**Note:** The Processing and Repricing statuses available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	ccrz__CartStatus__c	Picklist
Input or output with API <ul style="list-style-type: none"> <li>• With <b>SZ_S</b> in B2B Commerce for Visualforce Winter '21 (Winter 4.13) or later</li> <li>• With <b>SZ_M</b> in B2B Commerce for Visualforce Spring '20 (version 4.12) or earlier</li> </ul>	cartStatus	String

**Cart Type**

Whether the cart is a standard cart or a wish list.



Usage	Field Name	Data Type
Input on object	ccrz__CartType__c	Picklist
Input or output with API (with <b>SZ_M</b> or larger)	cartStatus	String

**CCEmailAddress**

Whitespace-separated list of email addresses for the CC line of the order confirmation email. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_page\\_checkout\\_cc\\_email.htm](https://help.salesforce.com/articleView?id=b2b_commerce_page_checkout_cc_email.htm).



**Tip:** We recommend that you populate this field only for orders that contain a low volume of cart items.

Usage	Field Name	Data Type
Input on object	ccrz__CCEmailAddress__c	Text Area (Long) (32768)
Input or output with API (with <b>SZ_XL</b> )	CCEmailAddress	String

**Contact**

The contact associated with the user assigned in the `User` field.

Usage	Field Name	Data Type
Input on object	ccrz__Contact__c	Lookup Relationship ( <a href="#">Contact</a> )
Input or output with API (with <b>SZ_XL</b> )	contact	ID

**Contract ID**

The contract ID when the cart's payment method is Contract. For more information about contract pricing, see [https://help.salesforce.com/articleView?id=sf.b2b\\_commerce\\_contract.htm](https://help.salesforce.com/articleView?id=sf.b2b_commerce_contract.htm).

Usage	Field Name	Data Type
Input on object	ccrz__ContractId__c	Text (50)
Input or output with API (with <b>SZ_XL</b> )	contractId	String

**Currency ISO Code**

The ISO 4217 currency code for the cart, such as *USD* or *JPY*.

Usage	Field Name	Data Type
Input on object	ccrz__CurrencyISOCode__c	Text (10)
Input or output with API (with <b>SZ_L</b> or larger)	currencyISOCode	String

**Effective Account ID**

The effective account associated with the cart. For usage, see

[https://help.salesforce.com/articleView?id=b2b\\_commerce\\_effective\\_account\\_cart.htm](https://help.salesforce.com/articleView?id=b2b_commerce_effective_account_cart.htm).

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveAccountID__c	Text (20)
Input or output with API <ul style="list-style-type: none"> <li>• With <b>SZ_M</b> in B2B Commerce for Visualforce Spring '20 (version 4.12) or later</li> <li>• With <b>SZ_XL</b> in B2B Commerce for Visualforce Summer '19 (version 4.11) or earlier</li> </ul>	effectiveAccountID	String

**Email**

Buyer's email address.

Usage	Field Name	Data Type
Input on object	ccrz__BuyerEmail__c	Email
Input or output with API (with <b>SZ_XL</b> )	buyerEmail	String

**Encrypted ID**

Semi-random UUID to refer to the cart in page URLs.

Usage	Field Name	Data Type
Input on object	ccrz__EncryptedId__c	Text (250)
Input or output with API (with <b>SZ_S</b> or larger)	encryptedId	String

**First Name**

Buyer's first name.

Usage	Field Name	Data Type
Input on object	ccrz__BuyerFirstName__c	Text (40)
Input or output with API (with <b>SZ_XL</b> )	buyerFirstName	String

**Last Name**

Buyer's last name.

Usage	Field Name	Data Type
Input on object	ccrz__BuyerLastName__c	Text (80)
Input or output with API (with <b>SZ_XL</b> )	buyerLastName	String

**Mobile Phone**

Buyer's mobile phone number.

Usage	Field Name	Data Type
Input on object	ccrz__BuyerMobilePhone__c	Phone
Input or output with API (with <b>SZ_XL</b> )	buyerMobilePhone	String

**Name**

A name for the cart in the My Carts section of the My Account page.

Usage	Field Name	Data Type
Input on object	ccrz__Name__c	Text (255)
Input or output with API (with <b>SZ_XL</b> )	name	String

**Note**

A shipping note entered during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__Note__c	Text Area (Long) (1000)
Input or output with API (with <b>SZ_XL</b> )	note	String

**Number of All Cart Items**


The total number of items in the cart, including both major and minor line items.

 **Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	ccrz__NumberOfAllCartItems__c	Roll-Up Summary (COUNT <a href="#">ccrz__E_CartItem__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	numberOfAllCartItems	Integer

**Number of Major Cart Items**

The total number of major line items in the cart.

 **Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	ccrz__NumberOfMajorCartItems__c	Roll-Up Summary (COUNT <a href="#">ccrz__E_CartItem__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	numberOfMajorCartItems	Integer

#### Payment Method

The payment method for the cart. This value corresponds to the value of the `ccrz__Payment_Method__c` field on the `ccrz__E_PaymentDetail__c` record.

Usage	Field Name	Data Type
Input on object	ccrz__PaymentMethod__c	Text (50)
Input or output with API (with <b>SZ_L</b> or larger)	paymentMethod	String

#### Phone

Buyer's phone number.

Usage	Field Name	Data Type
Input on object	ccrz__BuyerPhone__c	Phone
Input or output with API (with <b>SZ_XL</b> )	buyerPhone	String

#### PO Number

The purchase order (PO) number entered during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__PONumber__c	Text (50)
Input or output with API (with <b>SZ_L</b> or larger)	PONumber	String

#### Repriced Date

The date and time when B2B Commerce most recently priced cart items. By default, this value is the date and time the cart was created. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_page\\_shopping\\_cart\\_price.htm](https://help.salesforce.com/articleView?id=b2b_commerce_page_shopping_cart_price.htm).

 **Note:** This field is available only in B2B Commerce for Visualforce Spring '19 (version 4.10) or later.

Usage	Field Name	Data Type
Input on object	ccrz__RepricedDate__c	Date/Time
Input or output with API <ul style="list-style-type: none"> <li>With <b>SZ_M</b> in B2B Commerce for Visualforce Winter '21 (Winter 4.13) or later</li> <li>With <b>SZ_XL</b> in B2B Commerce for Visualforce Spring '20 (version 4.12) or earlier</li> </ul>	repricedDate	String

**Request Date**

The delivery date that the buyer requested during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__RequestDate__c	Date
Input or output with API (with <b>SZ_XL</b> )	requestDate	String

**Ship Amount**

Amount calculated for shipping the order during checkout. This value is calculated based on shipping rates and product quantities in the cart, and passed to the corresponding `ccrz__E_Order__c` record. For more information, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_shipping.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_shipping.htm).

Usage	Field Name	Data Type
Input on object	ccrz__ShipAmount__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	shipAmount	Decimal

**Ship Complete**

Whether the buyer selected the Ship Complete option during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__ShipComplete__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	shipComplete	Boolean

**Ship Discount Amount**

Amount to adjust the shipping cost by after the order is placed. This amount is specified on the cart, and passed to the corresponding `ccrz__E_Order__c` record. You can use your own back-end processing to apply this adjustment without changing the values of other fields.

Usage	Field Name	Data Type
Input on object	ccrz__ShipDiscountAmount__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	shipDiscountAmount	Decimal

**Ship Method**

The shipping method selected during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__ShipMethod__c	Text (50)
Input or output with API (with <b>SZ_XL</b> )	shipMethod	String

**Ship Structure**

The checkout experience used for grouping the cart's items for shipment. If your storefront uses line-level independence (LLI), this value is LLI. Otherwise, this value is empty.

Usage	Field Name	Data Type
Input on object	ccrz__ShipStructure__c	Text (128)
Input or output with API (with <b>SZ_XL</b> )	shipStructure	String

**Ship To**

The shipping address associated with the cart. This record is a unique copy of the shipping address specified during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__ShipTo__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with <b>SZ_XL</b> )	shipTo	ID

**Storefront**

The storefront where the cart was created.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with <b>SZ_XL</b> )	storefront	String

**Subtotal Amount**

The total entitled price of all items in the cart.

Usage	Field Name	Data Type
Input on object	ccrz__SubtotalAmount__c	Roll-Up Summary (SUM <a href="#">ccrz__E_CartItem__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	subtotalAmount	Decimal

**Tax Amount**

Tax amount calculated during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__TaxAmount__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	taxAmount	Decimal

**Tax Subtotal Amount**

The total taxable price of all items in the cart.

Usage	Field Name	Data Type
Input on object	ccrz__TaxSubTotalAmount__c	Roll-Up Summary (SUM <a href="#">ccrz__E_CartItem__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	taxSubTotalAmount	Decimal

**TaxExemptFlag**

Whether the cart is exempt from tax. This value is specified on the cart, and passed to the corresponding [ccrz\\_\\_E\\_Order\\_\\_c](#) record.

Usage	Field Name	Data Type
Input on object	ccrz__TaxExemptFlag__c	Checkbox
Input or output with API (with <b>SZ_L</b> or larger)	taxExemptFlag	Boolean

**Total Amount**

Total cost calculated for the cart, including:

- Ship amount
- Ship discount amount
- Subtotal amount
- Tax amount
- Adjustment amount

- Total surcharge

Usage	Field Name	Data Type
Input on object	ccrz__TotalAmount__c	Formula (Currency)
Input or output with API (with <b>SZ_L</b> or larger)	totalAmount	Decimal

#### Total Discount

Total amount discounted from the cart.

Usage	Field Name	Data Type
Input on object	ccrz__TotalDiscount__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	totalDiscount	Decimal

#### Total Quantity

Total number of items in the cart.

Usage	Field Name	Data Type
Input on object	ccrz__TotalQuantity__c	Roll-Up Summary (SUM <a href="#">ccrz__E_CartItem__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	totalQuantity	Integers

#### Total Surcharge

Charges other than taxes or shipping applied to the cart.

Usage	Field Name	Data Type
Input on object	ccrz__TotalSurcharge__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	totalSurcharge	Decimal

#### User

The user that created the cart.

Usage	Field Name	Data Type
Input on object	ccrz__User__c	Lookup Relationship ( <a href="#">User</a> )
Input or output with API (with <b>SZ_XL</b> )	user	ID



**Validation Status**

The cart's validation status, used with Apex remote actions.

Usage	Field Name	Data Type
Input on object	ccrz__ValidationStatus__c	Text (75)
Input or output with API (with <b>SZ_S</b> )	validationStatus	String

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Company Name	ccrz__BuyerCompanyName__c	Text (80)
Session ID	ccrz__SessionId__c	Text (255)

**Related Lists**

A record of the `ccrz__E_Cart__c` object can have multiple related lists that relate the cart to other types of records, such as cart items. The `ccrz.ccApiCart.fetch` method can return data from these related records, depending on the requested data size in the input map.

The reference for each related list indicates which data size the list is returned with by default. You can also use the [ccrz.ccApi.SZ\\_REL flag in your input map](#) to request a particular list, regardless of requested data size:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XS,
        ccrz.ccApi.SZ_REL => New List<String>{'E_CartItems__r'}
    }
}
```

**E\_CartItems**

List<Map<String, Object>> of `ccrz__E_CartItem__c`.

Consideration	Value
Default data size returned with	<b>SZ_S</b> or larger
<code>ccrz.ccApi.SZ_REL</code> request value	<code>E_CartItems__r</code>

**SEE ALSO:**

- [Salesforce Help: Considerations for Supporting a Buyer's Cart Experience](#)
- [Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)
- [Salesforce Help Scale Your Cart Capabilities for a Large Number of Items](#)
- [Salesforce Help Scaled Cart Page](#)

## ccrz\_\_E\_CartItem\_\_c

Represents an item in the buyer's shopping cart, including major and minor line items or applied coupons.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

Class used with: `ccrz.ccApiCart`

Fetch sizing: To specify which fields the `ccrz.ccApiCart.fetch` or `ccrz.ccApiCart.fetchCartItems` method returns for each cart item record, [request a specific data size](#) for the `ccrz.ccApiCart.CLI_ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.CLI_ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

### Fields

#### Absolute Discount

A fixed discount amount for the item in the cart's currency.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AbsoluteDiscount__c</code>	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	<code>absoluteDiscount</code>	Decimal

#### Adjustment Amount

Amount to adjust the item's price in the cart by. Typically, a customer service representative (CSR) performs this adjustment on behalf of a buyer.

Usage	Field Name	Data Type
Input on object	ccrz__AdjustmentAmount__c	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	adjustmentAmount	Decimal

**Availability Message**

Inventory availability information for the product. This value is copied from the `ccrz__AvailabilityMessageRT__c` field on the corresponding `ccrz__E_ProductInventoryItem__c` record.

Usage	Field Name	Data Type
Input on object	ccrz__AvailabilityMessage__c	Text Area (255)
Input or output with API (with <b>SZ_XL</b> )	availabilityMessage	String

**Cart**

The cart that the items belong to.

Usage	Field Name	Data Type
Input on object	ccrz__Cart__c	Master-Detail Relationship ( <code>ccrz__E_Cart__c</code> on page 25)
Input or output with API (with <b>SZ_S</b> or larger)	cart	ID
	cartR	Map<String, Object>, which contains a <code>sfdcName</code> field that references the cart's auto number.

**Cart Item Group**

The shipping group that the item belongs to. This value is set only when your storefront uses line-level independence (LLI) for checkout.

Usage	Field Name	Data Type
Input on object	ccrz__CartItemGroup__c	Lookup Relationship ( <code>ccrz__E_CartItemGroup__c</code> )
Input or output with API (with <b>SZ_XL</b> )	cartItemGroup	ID

**Cart Item ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__CartItemId__c	Text (128)
Input or output with API (with <b>SZ_XL</b> )	cartItemId	ID

**Cart Item Type**

Whether the cart item is a:

- Major line item, such as a standard product or a composite product
- Minor line item, such as a component product of a composite product
- Coupon

Usage	Field Name	Data Type
Input on object	ccrz__cartItemType__c	Picklist
Input or output with API (with <b>SZ_M</b> or larger)	cartItemType	String


**CartItemAutoNumber**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Category**

The category that the cart item's corresponding product belongs to.

 **Note:** The managed package doesn't set this field when building a cart, but the `ccrz.ccLogicCartPlace` logic service provider class copies this field's value to the corresponding `ccrz__E_OrderItem__c`.

Usage	Field Name	Data Type
Input on object	ccrz__Category__c	Lookup Relationship ( <a href="#">ccrz__E_Category__c</a> )
Input or output with API (with <b>SZ_XL</b> )	category	ID

**Comments**

Text that a buyer adds to the cart item on the Shopping Cart page.

Usage	Field Name	Data Type
Input on object	ccrz__Comments__c	Text Area (255)
Input or output with API (with <b>SZ_XL</b> )	comments	String

**Coupon**

The coupon record that corresponds to a coupon cart item.

Usage	Field Name	Data Type
Input on object	ccrz__Coupon__c	Lookup Relationship ( <a href="#">ccrz__E_Coupon__c</a> )
Input or output with API (with <b>SZ_XL</b> )	coupon	ID

**Display Product**

The product that displays on the Shopping Cart page and links to the Product Detail page for the cart item.

Usage	Field Name	Data Type
Input on object	ccrz__DisplayProduct__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	displayProduct	ID

**External Name**

The name for an externally priced product. This name displays for the cart item on the Shopping Cart page instead of the referenced display product's name.

Usage	Field Name	Data Type
Input on object	ccrz__ExtName__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	extName	String

**External SKU**

The SKU for an externally priced product. This SKU displays for the cart item on the Shopping Cart page instead of the referenced display product's SKU.

Usage	Field Name	Data Type
Input on object	ccrz__ExtSKU__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	extSKU	String

**Item Label**

A custom value to show for the item on the Shopping Cart page instead of the product name.

Usage	Field Name	Data Type
Input on object	ccrz__ItemLabel__c	Text (80)
Input or output with API (with <b>SZ_L</b> or larger)	itemLabel	String

**Item Status**

The cart item's status, based on specified product inventory rules:

- `Available`, if inventory is enabled and sufficient inventory is available for the product
- `Pre-Order`, if inventory is enabled but sufficient inventory isn't available for the product
- `Invalid` for all other cases

**Note:**

- For more information about inventory, see [Manage Product Inventory Items and Availability Messages](#).
- In B2B Commerce for Visualforce Winter '21 (version 4.13) or later, this value is `Invalid` when the corresponding `ccrz__E_Product__c` record's `ccrz__ProductStatus__c` field is any value other than `Released`. A value of `Invalid` causes the Scaled Cart page to show the product as unavailable in the cart. For more information, see [Unavailable Products on the Scaled Cart Page](#).

Usage	Field Name	Data Type
Input on object	<code>ccrz__ItemStatus__c</code>	Picklist
Input or output with API (with <b>SZ_L</b> or larger)	<code>itemStatus</code>	String

**Item Total**

Total cost calculated for the cart item, including:

- Subtotal amount
- Adjustment amount

Usage	Field Name	Data Type
Input on object	<code>ccrz__ItemTotal__c</code>	Formula (Currency)
Input or output with API (with <b>SZ_M</b> or larger)	<code>itemTotal</code>	Decimal

**Original Item Price**

The buyer's entitled price for the cart item before any adjustments are applied.

Usage	Field Name	Data Type
Input on object	<code>ccrz__OriginalItemPrice__c</code>	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	<code>originalItemPrice</code>	Decimal

**Original Quantity**

The original quantity per unit for component products included in kits, bundles, or dynamic kits.

Usage	Field Name	Data Type
Input on object	ccrz__OriginalQuantity__c	Number (16, 2)
Input or output with API (with <b>SZ_XL</b> )	originalQuantity	Integer

#### Parent Cart Item

The cart item that represents a composite product: a kit, bundle, or dynamic kit for which the current cart item is a component product.

Usage	Field Name	Data Type
Input on object	ccrz__ParentCartItem__c	Lookup Relationship (ccrz__E_CartItem__c)
Input or output with API (with <b>SZ_L</b> or larger)	parentCartItem	ID

#### Parent Product

By default, the managed package doesn't set this field.

 **Note:** The managed package doesn't set this field when building a cart, but the `ccrz.ccLogicCartPlace` logic service provider class copies this field's value to the corresponding `ccrz__E_OrderItem__c`.

Usage	Field Name	Data Type
Input on object	ccrz__ParentProduct__c	Lookup Relationship (ccrz__E_Product__c)
Input or output with API (with <b>SZ_XL</b> )	parentProduct	ID

#### Percent Discount

The percentage discount for this cart item when a coupon is applied.

Usage	Field Name	Data Type
Input on object	ccrz__PercentDiscount__c	Percent (3, 4)
Input or output with API (with <b>SZ_XL</b> )	percentDiscount	Decimal


#### Price

The buyer's entitled price for the cart item.

Usage	Field Name	Data Type
Input on object	ccrz__Price__c	Currency (16, 2)
Input or output with API (with <b>SZ_M</b> or larger)	price	Decimal

**Pricing Modifier Types**

Comma-delimited list of pricing modifiers applied to the cart item. The only value that the managed package supports by default is *tiered*. For more information about tiered pricing, see [Adjust Product Prices Based on Tiered Quantities](#).

 **Tip:** Avoid using the value *tiered* in any customizations or extensions in your subscriber code.

Usage	Field Name	Data Type
Input on object	ccrz__PricingModifierTypes__c	Text (255)
Input or output with API (with <b>SZ_XL</b> )	pricingModifierTypes	String

**Pricing Type**

An internal value that indicates the cart item's pricing source.

- When this value is *ATTRGROUP*, B2B Commerce for Visualforce evaluates the cart item as part of an attribute-based product.
- When this value is *external*, B2B Commerce for Visualforce evaluates the cart item as externally priced.

Usage	Field Name	Data Type
Input on object	ccrz__PricingType__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	pricingType	String

**Primary Attribute**

The primary attribute selected on the cart item's corresponding attribute-based product.

Usage	Field Name	Data Type
Input on object	ccrz__PrimaryAttr__c	Lookup Relationship ( <a href="#">ccrz__E_Attribute__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	primaryAttr	ID
	primaryAttrR	Map<String, Object>, which contains a <code>parentAttribute</code> field that references the primary attribute's parent attribute.

**Product**

The product that corresponds to the cart item.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	product	ID



Usage	Field Name	Data Type
	productR	Map<String, Object>, which contains the following fields: <ul style="list-style-type: none"> <li>• endDate</li> <li>• productStatus</li> <li>• productType</li> <li>• SKU</li> <li>• startDate</li> </ul>

**ProductType**

The type of product that corresponds to the cart item. For supported values, see [Types of Products for B2B Commerce for Visualforce](#).

Usage	Field Name	Data Type
Input on object	ccrz__ProductType__c	Text (20)
Input or output with API (with <b>SZ_M</b> or larger)	productType	String

**Quantity**

The quantity for the item in the cart.

Usage	Field Name	Data Type
Input on object	ccrz__Quantity__c	Number (18, 0)
Input or output with API (with <b>SZ_S</b> or larger)	quantity	Integer

**Recurring Price**

The recurring price amount for the cart item's subscription.

Usage	Field Name	Data Type
Input on object	ccrz__RecurringPrice__c	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	recurringPrice	Decimal

**Recurring Price Sub Amount**


Recurring price subtotal for the total quantity of the cart item.

Usage	Field Name	Data Type
Input on object	ccrz__RecurringPriceSubAmt__c	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	recurringPriceSubAmt	Decimal

**Request Date**

The date when the buyer added the product to their shopping cart.

Usage	Field Name	Data Type
Input on object	ccrz__RequestDate__c	Date
Input or output with API (with <b>SZ_XL</b> )	requestDate	Date

 **Note:** By default, this field isn't synchronized with the `ccrz__RequestDate__c` field on the parent `ccrz__E_Cart__c` object. To implement this synchronization, extend the `ccrz.ccLogicCartAddTo.setCartItemFields` logic service provider method. In your extension, set the `ccrz__RequestDate__c` value on the `ccrz__E_CartItem__c` record to `null`, which causes the `ccrz__E_CartItem__c` record to inherit the `ccrz__RequestDate__c` value from the parent `ccrz__E_Cart__c` record. This value propagates to the corresponding `ccrz__E_OrderItem__c` when the buyer places an order from the cart.

```
global with sharing class customLogicCartAddTo extends ccrz.ccLogicCartAddTo {
    global override Map<String, Object> setCartItemFields(Map<String, Object>
inputData);
        ccrz__E_CartItem__c cartItem = (ccrz__E_CartItem__c)
inputData;
    }
}
```

**Secondary Attribute**

The secondary attribute selected on the cart item's corresponding attribute-based product.

Usage	Field Name	Data Type
Input on object	ccrz__SecondAttr__c	Lookup Relationship ( <code>ccrz__E_Attribute__c</code> )
Input or output with API (with <b>SZ_S</b> or larger)	secondaryAttr	ID
	secondaryAttrR	Map<String, Object>, which contains a <code>parentAttribute</code> field that references the secondary attribute's parent attribute.

**Selected Attributes**

JSON content that describes the selected attributes for the cart item's corresponding product.

Usage	Field Name	Data Type
Input on object	ccrz__SelectedAttr__c	Text Area (Long) (1000)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_M</b> or larger)	selectedAttr	ID


**Seller**

The seller that the buyer is purchasing the product from.

Usage	Field Name	Data Type
Input on object	ccrz__Seller__c	Lookup Relationship ( <a href="#">ccrz__E_Seller__c</a> )
Input or output with API (with <b>SZ_XL</b> )	seller	ID
	sellerR	Map<String, Object>, which contains a ID field that references the seller ID.

**Sort Name**

The localized product name for sorting the cart item on the Shopping Cart page when Cart Scale is enabled. For usage, see [Options for Buyers to Sort and Search Items on the Scaled Cart Page](#).

 **Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	ccrz__SortName__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	sortName	String

**Sort Price**

The reference price for sorting the cart item on the Shopping Cart page when Cart Scale is enabled.

 **Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	ccrz__SortPrice__c	Currency (16, 2)
Input or output with API (with <b>SZ_M</b> or larger)	sortPrice	Decimal

**Store ID**

The storefront where the cart was created.

Usage	Field Name	Data Type
Input on object	ccrz__StoreID__c	Text (128)
Input or output with API (with <b>SZ_M</b> or larger)	storeId	String

**Sub Amount**

The entitled subtotal price for the total quantity of the cart item.

Usage	Field Name	Data Type
Input on object	ccrz__SubAmount__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	subAmount	Decimal

**SubProdTerm**

The subscription that corresponds to the cart item.

Usage	Field Name	Data Type
Input on object	ccrz__SubProdTerm__c	Lookup Relationship ( <a href="#">ccrz__E_SubProdTerm__c</a> )
Input or output with API (with <b>SZ_XL</b> )	subProdTerm	ID

**Tertiary Attribute**

The tertiary attribute selected on the cart item's corresponding attribute-based product.

Usage	Field Name	Data Type
Input on object	ccrz__TertiaryAttr__c	Lookup Relationship ( <a href="#">ccrz__E_Attribute__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	tertiaryAttr	ID
	tertiaryAttrR	Map<String, Object>, which contains a parentAttribute field that references the tertiary attribute's parent attribute.

**Unit Of Measure**

The unit of measure from the cart item's corresponding product.

Usage	Field Name	Data Type
Input on object	ccrz__UnitOfMeasure__c	Text (32)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_L</b> or larger)	unitOfMeasure	String

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Is Subscription Selected	ccrz__Is_Subscription_Selected__c	Checkbox
Subscription Duration (In Months)	ccrz__Subscription_Duration__c	Number (18, 0)
Subscription Frequency	ccrz__Subscription_Frequency__c	Number (4, 0)

SEE ALSO:

- [Salesforce Help Types of Products for B2B Commerce for Visualforce](#)
- [Salesforce Help Offer Coupons for Discounts on Storefront Purchases](#)
- [Salesforce Help Scale Your Cart Capabilities for a Large Number of Items](#)
- [Salesforce Help Scaled Cart Page](#)
- [Salesforce Help Set Product Prices in the Cart from an External Source](#)

## ccrz\_\_E\_CartItemGroup\_\_c

Represents a shipping group of cart items for line-level independence (LLI) checkout.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

### Methods used with:

- [ccrz.ccApiCart.createCartItemGroup](#)
- [ccrz.ccApiCart.initItemGroups](#)
- [ccrz.ccApiCart.reviseCartItemGroup](#)

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request](#) in your [input map](#).

## Fields

### CartItemGroupId

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CartItemGroupId__c</code>	Text (128)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### CC Cart

The cart where the shipping group applies.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Cart__c</code>	Master-Detail Relationship ( <a href="#">ccrz__E_Cart__c</a> on page 25)
Input or output with API (with any data size)	<code>cart</code>	String

### Group Name

The shipping group name, used for organizing groups on the Checkout page.

Usage	Field Name	Data Type
Input on object	<code>ccrz__GroupName__c</code>	Text (255)
Input or output with API (with any data size)	<code>groupName</code>	String

### Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	<code>Name</code>	Auto Number

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Note**

Shipping notes for the group.

Usage	Field Name	Data Type
Input on object	ccrz__Note__c	Text Area (Long) (1,000)
Input or output with API (with any data size)	note	String

**Request Date**

The buyer's requested delivery date for the shipping group.

Usage	Field Name	Data Type
Input on object	ccrz__RequestDate__c	Date
Input or output with API (with any data size)	requestDate	Date

**Ship Amount**

The shipping cost calculated for the group, based on shipping rates and product quantities in the group. This value is passed to the corresponding [ccrz\\_\\_E\\_Cart\\_\\_c](#) and [ccrz\\_\\_E\\_Order\\_\\_c](#) records. For more information, see [Specify Shipping Rates for a Product](#).

Usage	Field Name	Data Type
Input on object	ccrz__ShipAmount__c	Currency (16, 2)
Input or output with API (with any data size)	shipAmount	Decimal

**Ship Method**

The shipping method for the group.

Usage	Field Name	Data Type
Input on object	ccrz__ShipMethod__c	Text (50)
Input or output with API (with any data size)	shipMethod	String

**Ship To**

The shipping address for the group.

Usage	Field Name	Data Type
Input on object	ccrz__ShipTo__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with any data size)	shipTo	ID

**ccrz\_\_E\_CartCoupon\_\_c**

Represents the junction of a coupon and a shopping cart record.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**API Usage**

**Class used with:** The API doesn't directly query records of this object.

Instead, the [ccrz.ccApiCoupon](#) return data includes the `E_CartCouponsS` related list when at least one coupon has been applied to the cart.

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

**Fields****Cart**

The cart where the coupon is applied.

Usage	Field Name	Data Type
Input on object	ccrz__Cart__c	Master-Detail Relationship ( <a href="#">ccrz__E_Cart__c</a> on page 25)



Usage	Field Name	Data Type
Input or output with API (with any data size)	cart	ID

**CartCoupon ID**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	ccrz__CartCouponId__c	Auto Number
Input or output with API (with any data size)	cartCouponId	String

**Coupon**

The coupon that's applied to the cart.

Usage	Field Name	Data Type
Input on object	ccrz__Coupon__c	Master-Detail Relationship ( <a href="#">ccrz__E_Coupon__c</a> on page 82)
Input or output with API (with any data size)	coupon	ID

**E\_CartCoupon Name**

A unique name for this record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfdcName	String

**Enabled**

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_Coupon\\_\\_c](#)

[Salesforce Help Offer Coupons for Discounts on Storefront Purchases](#)

## ccrz\_\_E\_Category\_\_c

Represents a category record on the storefront.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** [ccrz.ccApiCategory](#)

**Fetch sizing:** To specify which fields the [ccrz.ccApiCategory.fetch](#) method returns for each category record, [request a specific data size](#) for the `ccrz.ccApiCart.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

### Fields

#### Category ID

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__CategoryID__c	Text (128)
Input or output with API (with <b>SZ_S</b> or larger)	categoryID	String

**Category Name**

The name to show on the storefront for the category.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Child Category Count**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__ChildCategoryCount__c	Number (4, 0)
Input or output with API (with <b>SZ_L</b> or larger)	childCategoryCount	String

**Data ID**

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	ccrz__DataId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**End Date**

The last day when the category is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with <b>SZ_XL</b> )	endDate	Date

**Long Description**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__LongDescRT__c	Text Area (Rich) (5000)
Input or output with API (with <b>SZ_M</b> or larger)	longDesc	String

**Parent Category**

The parent category for the current category.

Usage	Field Name	Data Type
Input on object	ccrz__ParentCategory__c	Lookup Relationship ( <a href="#">ccrz__E_Category__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	parentCategory	ID

**Product Count**

Formula that calculates the number of products assigned to the current category.

Usage	Field Name	Data Type
Input on object	ccrz__ProductCount__c	Roll-Up Summary (COUNT <a href="#">ccrz__E_ProductCategory__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	productCount	Integer

**SEO ID**

A [unique, URL-safe string](#) (up to 255 characters) for the category's friendly URL path. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_seo\\_product\\_category.htm](https://help.salesforce.com/articleView?id=b2b_commerce_seo_product_category.htm).



**Tip:** The SEO ID field doesn't support the following special characters:

!	*	'	(	)	@	&
;	:	+	=	\$	,	.
/	?	%	#	[	]	

Usage	Field Name	Data Type
Input on object	ccrz__SEOId__c	Text (255)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_S</b> or larger)	SEOId	String

**Sequence**

Order of the category under the parent category, relative to other categories assigned to the same parent category. Lower values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with <b>SZ_M</b> or larger)	sequence	Integer

**Short Description**

The description for the category that displays on the Product List page. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_page\\_product\\_list\\_subcategory.htm](https://help.salesforce.com/articleView?id=b2b_commerce_page_product_list_subcategory.htm).

Usage	Field Name	Data Type
Input on object	ccrz__ShortDescRT__c	Text Area (Rich) (2000)
Input or output with API (with <b>SZ_M</b> or larger)	shortDesc	String

**Start Date**

The first day when the category is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with <b>SZ_XL</b> )	startDate	Date

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Long Desc	ccrz__LongDesc__c	Text Area (Long) (1000)
Short Desc	ccrz__ShortDesc__c	Text (255)

## Related Lists

A record of the `ccrz__E_Category__c` object can have multiple related lists that relate the category to other types of records, such as product assignments or category media. The `ccrz.ccApiCategory.fetch` method can return data from these related records, depending on the requested data size in the input map.

The reference for each related list indicates which data size the list is returned with by default. You can also use the [ccrz.ccApi.SZ\\_REL flag in your input map](#) to request a particular list, regardless of requested data size:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
    ccrz.ccApi.SZ_REL => New List<String>{'Category_Medias__r'}
  }
}
```

### categoryMediasS

List<Map<String, Object>> of `ccrz__E_CategoryMedia__c`.

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger
<code>ccrz.ccApi.SZ_REL</code> request value	Category_Medias__r

### productCategoriesS

List<Map<String, Object>> of `ccrz__E_ProductCategory__c`.

Consideration	Value
Default data size returned with	<b>SZ_M</b> or larger
<code>ccrz.ccApi.SZ_REL</code> request value	Product_Categories__r

SEE ALSO:

[Salesforce Help Organize Products into Categories for B2B Commerce for Visualforce](#)

[Salesforce Help Specify SEO Metadata for Products and Categories](#)

## ccrz\_\_E\_Category|18N\_\_c

Represents an internationalized category record for a specific locale.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

The API doesn't return fields on the `ccrz__E_CategoryI18N__c` record directly. When the `ccrz.ccApiCategory.fetch` method's input map includes the `ccrz.ccApiCategory.CATEGORYLOCALE` key, the method:

1. References `ccrz__E_CategoryI18N__c` records for the specified locale.
2. Returns `ccrz__E_Category__c` records, translated with the localized data.

## Fields

### Category

The category record to localize.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Category__c</code>	Master-Detail Relationship ( <code>ccrz__E_Category__c</code> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Category I18N Ext Id

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CategoryI18NId__c</code>	Text (128)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Category I18N Name

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	Provides the value for the <code>sfdcName</code> field when the API queries localized category records.	

**Data ID**

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__DataId__c</code>	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Locale**

The [storefront locale](#) that the category is localized for.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Locale__c</code>	Text (25)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**LongDescRT**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__LongDescRT__c</code>	Text Area (Rich) (5000)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**SEO ID**

The localized [unique, URL-safe string](#) (up to 255 characters) for the category's friendly URL path. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_seo\\_product\\_category.htm](https://help.salesforce.com/articleView?id=b2b_commerce_seo_product_category.htm).

 **Tip:** The SEO ID field doesn't support the following special characters:

!	*	'	(	)	@	&
;	:	+	=	\$	,	.
/	?	%	#	[	]	

Usage	Field Name	Data Type
Input on object	<code>ccrz__SEOId__c</code>	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	



**ShortDescRT**

The localized description for the category that displays on the Product List page. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_page\\_product\\_list\\_subcategory.htm](https://help.salesforce.com/articleView?id=b2b_commerce_page_product_list_subcategory.htm).

Usage	Field Name	Data Type
Input on object	ccrz__ShortDescRT__c	Text Area (Rich) (2000)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Long Desc	ccrz__LongDesc__c	Text Area (Long) (5000)
Short Desc	ccrz__ShortDesc__c	Text (2000)

SEE ALSO:

[Salesforce Help Localize Categories for an Internationalized Storefront](#)

**ccrz\_\_E\_CategoryMedia\_\_c**

Represents an image for a category that appears on the Product List page.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**API Usage**

**Class used with:** [ccrz.ccApiCategory](#)

**Fetch sizing:** The `ccrz.ccApiCategory.fetch` method returns the `categoryMediasS` related list when you [request at least a large data size](#) for the `ccrz.ccApiCategory.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
  }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
    ccrz.ccApi.SZ_REL => New List<String>{'Category_Medias__r'}
  }
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

### AltMessage

The value of an image's `alt` attribute in the page HTML. For usage, see [Guidelines for Category Images for B2B Commerce for Visualforce](#).

Usage	Field Name	Data Type
Input on object	<code>ccrz__AltMessage__c</code>	Text (100)
Input or output with API (with any data size)	<code>altMessage</code>	String

### Category Media Ext Id

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CategoryMediaId__c</code>	Text (128)
Input or output with API (with any data size)	<code>categoryMediaId</code>	String

### Category

The category that the image applies to.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Category__c</code>	Master-Detail Relationship ( <a href="#">ccrz__E_Category__c</a> )

Usage	Field Name	Data Type
Input or output with API (with any data size)	category	ID

**Category Media Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

**Category Media Source**

Whether the source for the image is a static resource, content delivery network (CDN), or file attachment. For usage, see:

- [Guidelines for Category Images for B2B Commerce for Visualforce](#)
- [Source Options for Storefront Media](#)

Usage	Field Name	Data Type
Input on object	ccrz__ProductMediaSource__c	Picklist
Input or output with API (with any data size)	productMediaSource	String

**Enabled**

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	enabled	Boolean

**End Date**

The last day when the category media is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with any data size)	endDate	Date

**File Path**

The path to the image file from the root of a static resource's ZIP file. For usage, see [Add a Category Image from a Static Resource](#).

Usage	Field Name	Data Type
Input on object	ccrz__FilePath__c	Text (128)
Input or output with API (with any data size)	filePath	String

**Locale**

The [storefront locale](#) that the image is localized for. For usage, see [Add a Category Image from a Static Resource](#).

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (10)
Input or output with API (with any data size)	locale	String

**Media Type**

Whether the media is a category image or category image thumbnail. For usage, see [Guidelines for Category Images for B2B Commerce for Visualforce](#).

Usage	Field Name	Data Type
Input on object	ccrz__MediaType__c	Picklist
Input or output with API (with any data size)	mediaType	String

**Sequence**

The priority of the image for the category. Only one category image or category image thumbnail displays for each category. The Product List page shows the image with the lowest Sequence value.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with any data size)	sequence	Integer

**Start Date**

The first day when the category media is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date

Usage	Field Name	Data Type
Input or output with API (with any data size)	startDate	Date

**Static Resource Name**

The name of the static resource that contains the image. For usage, see [Add a Category Image from a Static Resource](#).

Usage	Field Name	Data Type
Input on object	ccrz__StaticResourceName__c	Text (255)
Input or output with API (with any data size)	staticResourceName	String

**URI**

The fully qualified URL to the image file from a CDN. For usage, see [Add a Category Image from a Content Delivery Network](#).

Usage	Field Name	Data Type
Input on object	ccrz__URI__c	Text (255)
Input or output with API (with any data size)	uri	String

**SEE ALSO:**

[Salesforce Help Guidelines for Category Images for B2B Commerce for Visualforce](#)

**ccrz\_\_E\_CategorySpec\_\_c**

Represents a spec value that's assigned to all products in a category.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** The API doesn't directly query records of this object.

The `ccrz.ccApiCategory.fetch` method returns data for products that belong to the categories that the method queries. When you create a category spec, the returned product data includes the product spec value that the category spec record assigns.

## Fields

### Category

The category whose products the spec applies to.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Category__c</code>	Master-Detail Relationship ( <a href="#">ccrz__E_Category__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Category Spec Ext Id

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CategorySpecId__c</code>	Text (128)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Category Spec Number

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Spec

The specification to define for all the products in the category, such as Material.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Spec__c</code>	Master-Detail Relationship ( <a href="#">ccrz__E_Spec__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

**SpecValue**

Value of the product's spec, such as Stainless Steel.

**Tip:**

- The Spec Value field is case-insensitive. If the spec's filter type is Checkbox, the Feature Filtering widget considers any case variation of the same string to be the same spec value. For example, the values *Commercial*, *commercial*, and *COMMERCIAL* are all counted as the same value.
- If the spec's filter type is Slider:
  - Make sure that all spec values you create for that spec are numbers with consistent decimal places. For example, the range of values *1 . 5*, *2 . 0*, and *3* isn't valid—instead of *3*, enter *3 . 0*.
  - Avoid specifying negative numbers for any spec values, which can break the default slider functionality.

Usage	Field Name	Data Type
Input on object	ccrz__SpecValue__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

SEE ALSO:

[Salesforce Help Apply the Same Spec Values to All Products in a Category](#)

**ccrz\_\_E\_CompositeProduct\_\_c**

Represents composite and component product relationships for bundles, kits, dynamic kits, aggregated products, and assembly products.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**API Usage**

**Class used with:** [ccrz.ccApiProduct](#)

**Fetch sizing:** The `ccrz.ccApiProduct.fetch` method returns the `compositeProductsS` related list when you [request at least a large data size](#) for the `ccrz.ccApiProduct.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
  }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
    ccrz.ccApi.SZ_REL => New List<String>{'CompositeProducts__r'}
  }
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

### Component

Component of a composite product.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Component__c</code>	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	<code>component</code>	ID
	<code>componentR</code>	<code>Map&lt;String, Object&gt;</code> that describes the component product.

### Composite

The composite product.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Composite__c</code>	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	<code>composite</code>	ID

### Composite Product Ext Id

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CompositeProductId__c</code>	Text (128)



Usage	Field Name	Data Type
Input or output with API (with any data size)	compositeProductId	String

**Composite Product Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

**DescriptionRT**

Optional text for internal reference. This field isn't used when displaying composite products on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__DescriptionRT__c	Text Area (Rich) (4000)
Input or output with API (with any data size)	description	String

**Parent**

A composite product under which the current composite product is nested, such as when a bundle is included as a component of a dynamic kit.

Usage	Field Name	Data Type
Input on object	ccrz__Parent__c	Lookup Relationship (ccrz__E_CompositeProduct__c)
Input or output with API (with any data size)	parent	ID

**Product Group**

Related product group referenced from a dynamic kit's product guide. This field is used only when a composite product is a dynamic kit.

Usage	Field Name	Data Type
Input on object	ccrz__ProductGroup__c	Lookup Relationship (ccrz__E_RelatedProductGroup__c)
Input or output with API (with any data size)	productGroup	ID

Usage	Field Name	Data Type
	productGroupR	Map<String, Object> that describes the product group.

**Quantity**

The quantity of a component product included in a kit or bundle.

Usage	Field Name	Data Type
Input on object	ccrz__Quantity__c	Number (18, 0)
Input or output with API (with any data size)	quantity	Integer

**Sequence**

The order of a component product in a kit or bundle shown on the Product Detail page.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with any data size)	sequence	Integer

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Category	ccrz__Category__c	Lookup Relationship (ccrz__E_Category__c)
Description	ccrz__Description__c	Text Area (Long) (4000)
Extended Price	ccrz__ExtendedPrice__c	Currency (16, 2)

Field Label	Field Name	Data Type
Optional	ccrz__Optional__c	Checkbox

## SEE ALSO:

[Salesforce Help Types of Products for B2B Commerce for Visualforce](#)

[Salesforce Help Create a Bundle](#)

[Salesforce Help Create a Kit](#)

[Salesforce Help Create a Dynamic Kit](#)

[Salesforce Help Create an Aggregated Product](#)

[Salesforce Help Create an Assembly Product](#)

## ccrz\_\_E\_Configuration\_\_c

Represents a storefront or global configuration setting.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** [ccrz.ccApiConfig](#)

**Fetch sizing:** The [ccrz.ccApiConfig.fetch](#) method returns the `ccrz.ccApiConfig.CONFIGURATIONS` key, which contains records of this object. The key for each configuration record returned is `moduleAPIName.configurationAPIName`, such as `ct.enabled`.

The API doesn't return values from this object's fields.

### Fields

#### CC Configuration Metadata

The metadata record for this specific configuration.

Usage	Field Name	Data Type
Input on object	ccrz__ConfigurationMeta__c	Lookup Relationship (ccrz__E_ConfigurationMeta__c)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Configuration ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__ConfigurationId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Name**

The configuration's name, in the format *storefront.page.module.configuration*.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Page Name**

The page that the configuration applies to. This value is a [page key](#) defined in CC Admin global settings, such as *hp* for Home page and *p1p* for Product List page.

Usage	Field Name	Data Type
Input on object	ccrz__PageName__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Sequence Priority**

Internal use only. Don't modify the value that the managed package assigns to this field.

Usage	Field Name	Data Type
Input on object	ccrz__SequencePriority__c	Number (4, 0)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Storefront**

The storefront that the configuration applies to.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Text Value**

The configuration's value.

Usage	Field Name	Data Type
Input on object	ccrz__TextValue__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

SEE ALSO:

[Salesforce Help Manage Configuration Settings](#)

**ccrz\_\_E\_ConfigurationMeta\_\_c**

Represents metadata for an available storefront or global configuration setting.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**API Usage**

**Class used with:** The API doesn't directly query records of this object.

## Fields

### API Name

A name for programmatically referencing the configuration setting. If a configuration module contains multiple configurations, each configuration must have a unique API name. However, the same API name can apply to configurations across multiple modules.

For example, the Enabled configuration exists in the Breadcrumb, Header, Promotion, and many other modules. Each module can contain only one configuration with the API name `enabled`, but a configuration with the API name `enabled` can exist in multiple modules.

Usage	Field Name	Data Type
Input on object	<code>ccrz__APIName__c</code>	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### CC Configuration Module

The module that the configuration belongs to, such as Header or Product List.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ConfigurationModule__c</code>	Lookup Relationship ( <a href="#">ccrz__E_ConfigurationModule__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Configuration Meta ID

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ConfigurationMetaId__c</code>	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Description

Configuration description that's visible as a tooltip in CC Admin global and storefront configuration settings.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Description__c</code>	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Externally Safe**

Whether the configuration is safe to export to a system outside of the B2B Commerce managed package. For example, a configuration for showing the cart total price in the Header component is probably externally safe. But, a configuration that stores a credential for another system probably isn't externally safe.

Usage	Field Name	Data Type
Input on object	ccrz__ExternallySafe__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Name**

The name of the configuration, such as Enabled or Display SKU.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	The API doesn't return data for this field.	

SEE ALSO:

[Salesforce Help Manage Configuration Settings](#)

## ccrz\_\_E\_ConfigurationModule\_\_c

Represents the definition of a module that contains global or storefront configuration settings.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** The API doesn't directly query records of this object.

## Fields

### API Name

A name for programmatically referencing the configuration module and its settings. For example, the Category Tree module's API name is `ct`.

Usage	Field Name	Data Type
Input on object	<code>ccrz__APIName__c</code>	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Configuration Module ID

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ConfigurationModuleId__c</code>	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Name

The name of the module, such as Category Tree or Promotion.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	The API doesn't return data for this field.	

SEE ALSO:

[Salesforce Help Manage Configuration Settings](#)

## ccrz\_\_E\_ContactAddr\_\_c

Represents address details associated with carts, orders, invoices, subscriptions, payments, and other storefront data. A contact address is associated with an account address book, which references a specific account and specifies whether the address is for shipping or billing.

## Compatibility

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiAddress`

**Fetch sizing:** To specify which fields the `ccrz.ccApiAddress.fetch` method returns for each contact address record, [request a specific data size](#) for the `ccrz.ccApiAddress.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Address Firstline

First line of the street address.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AddressFirstline__c</code>	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	<code>addressFirstline</code>	String

### Address Secondline

Second line of the street address.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AddressSecondline__c</code>	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	<code>addressSecondline</code>	String

### Address Thirdline

Third line of the street address.

Usage	Field Name	Data Type
Input on object	ccrz__AddressThirdline__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	addressThirdline	String

**AddrReadOnly**

Whether a buyer can add, edit, or delete the address in the Manage Address Book section of the My Account page. For usage, see [Allow a Buyer to Manage Shipping and Billing Addresses on the My Account Page](#).

Usage	Field Name	Data Type
Input on object	ccrz__AddrReadOnly__c	Checkbox
Input or output with API (with <b>SZ_M</b> or larger)	addrReadOnly	Boolean

**City**

City for the address.

Usage	Field Name	Data Type
Input on object	ccrz__City__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	city	String

**Company Name**

Name of a company associated with this address.

Usage	Field Name	Data Type
Input on object	ccrz__CompanyName__c	Text (100)
Input or output with API (with <b>SZ_M</b> or larger)	companyName	String

**Contact Addr ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__ContactAddrId__c	Text (128)
Input or output with API (with <b>SZ_M</b> or larger)	contactAddrId	String

**Contact Auto Number**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	ID

**Country**

Name of the country or territory for the address.

Usage	Field Name	Data Type
Input on object	ccrz__Country__c	Text (50)
Input or output with API (with <b>SZ_M</b> or larger)	country	String

**Country ISO Code**

ISO 3166-1 code for the country or territory.

Usage	Field Name	Data Type
Input on object	ccrz__CountryISOCode__c	Text (10)
Input or output with API (with <b>SZ_M</b> or larger)	countryISOCode	String

**Daytime Phone**

Primary phone number for the address.

Usage	Field Name	Data Type
Input on object	ccrz__DaytimePhone__c	Phone
Input or output with API (with <b>SZ_M</b> or larger)	daytimePhone	String

**Email**

Email associated with the address.

Usage	Field Name	Data Type
Input on object	ccrz__Email__c	Email
Input or output with API (with <b>SZ_M</b> or larger)	email	String

**First Name**

First name of the person associated with the address.

Usage	Field Name	Data Type
Input on object	ccrz__FirstName__c	Text (100)
Input or output with API (with <b>SZ_M</b> or larger)	firstName	String

**Home Phone**

Alternative phone number for the address.

Usage	Field Name	Data Type
Input on object	ccrz__HomePhone__c	Phone
Input or output with API (with <b>SZ_M</b> or larger)	homePhone	String

**Last Name**

Last name of the person associated with the address.

Usage	Field Name	Data Type
Input on object	ccrz__LastName__c	Text (100)
Input or output with API (with <b>SZ_M</b> or larger)	lastName	String

**Mail Stop**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__MailStop__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	mailStop	String

**Middle Name**

Middle name of the person associated with the address.

Usage	Field Name	Data Type
Input on object	ccrz__MiddleName__c	Text (100)
Input or output with API (with <b>SZ_M</b> or larger)	middleName	String

**Partner ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__Partner_Id__c	Text (100)
Input or output with API (with <b>SZ_M</b> or larger)	partnerId	String

**Postal Code**

Postal code for the address.

Usage	Field Name	Data Type
Input on object	ccrz__PostalCode__c	Text (10)
Input or output with API (with <b>SZ_M</b> or larger)	postalCode	String

**Shipping Comments**

Optional shipping comments entered during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__ShippingComments__c	Text Area (Long) (32000)
Input or output with API (with <b>SZ_M</b> or larger)	shippingComments	String

**State**

Name of the state for an address that's located *outside* of the United States or Canada.

Usage	Field Name	Data Type
Input on object	ccrz__State__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	state	String

**State ISO Code**

ISO 3166-2 code for a US state or Canadian province.

Usage	Field Name	Data Type
Input on object	ccrz__StateISOCode__c	Text (10)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_M</b> or larger)	stateISOCode	String

SEE ALSO:

[ccrz\\_\\_E\\_AccountAddressBook\\_\\_c](#)

[Salesforce Help Account Address Books and Contact Addresses for B2B Commerce for Visualforce Buyers](#)

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

## ccrz\_\_E\_Coupon\_\_c

Represents a coupon record for the storefront.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** [ccrz.ccApiCoupon](#)

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

### Fields

#### Cart Total Amount

The total price of items in the shopping cart that must exist before the coupon is valid. For usage, see [Offer a Discount on Cart Total Price](#).

Usage	Field Name	Data Type
Input on object	ccrz__CartTotalAmount__c	Currency (16,2)
Input or output with API (with any data size)	cartTotalAmount	Decimal

**Coupon Auto Number**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

**Coupon Code**

The code that you want a buyer to enter on the Shopping Cart page. For usage, see [Guidelines When Creating Coupons for Your Storefront](#).

Usage	Field Name	Data Type
Input on object	ccrz__CouponCode__c	Text (20)
Input or output with API (with any data size)	couponCode	String

**Coupon ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__CouponId__c	Text (128)
Input or output with API (with any data size)	couponId	String

**Coupon Name**

The name that appears on the Shopping Cart page after a buyer applies the coupon. For usage, see [Guidelines When Creating Coupons for Your Storefront](#).

Usage	Field Name	Data Type
Input on object	ccrz__CouponName__c	Text (128)
Input or output with API (with any data size)	couponName	String

**Coupon Notes**

The managed package only uses this field internally.

Usage	Field Name	Data Type
Input on object	ccrz__Coupon_Notes__c	Text (255)

Usage	Field Name	Data Type
Input or output with API (with any data size)	couponNotes	String

**Currency ISO Code**

The ISO 4217 currency code for the coupon, such as *USD* or *JPY*. For usage, see [Guidelines When Creating Coupons for Your Storefront](#).

Usage	Field Name	Data Type
Input on object	ccrz__CurrencyISOCODE__c	Text (10)
Input or output with API (with any data size)	currencyISOCODE	String

**Discount Amount**

The coupon's fixed or percentage discount to apply. For usage, see [Guidelines When Creating Coupons for Your Storefront](#).

Usage	Field Name	Data Type
Input on object	ccrz__DiscountAmount__c	Currency (16, 2)
Input or output with API (with any data size)	discountAmount	Decimal

**Discount Type**

Whether the coupon's discount a fixed amount, percentage amount, or an override price. For usage, see [Guidelines When Creating Coupons for Your Storefront](#).

Usage	Field Name	Data Type
Input on object	ccrz__DiscountType__c	Text (30)
Input or output with API (with any data size)	discountType	String

**Enabled**

Whether the coupon is active on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	enabled	Boolean



**End Date**

The last day when the coupon is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with any data size)	endDate	Date

**Max Use**

The limit for the number of times that the coupon can be used considering all storefront users. For example, if you enter 300, the coupon can be applied up to 300 times across all carts and all users on your storefront. For usage, see [Guidelines When Creating Coupons for Your Storefront](#).

Usage	Field Name	Data Type
Input on object	ccrz__MaxUse__c	Number (18, 0)
Input or output with API (with any data size)	maxUse	Integer

**Minimum Quantity**

A specific number of products that must exist in the cart before the coupon is valid. For usage, see [Guidelines When Creating Coupons for Your Storefront](#).

Usage	Field Name	Data Type
Input on object	ccrz__MinQty__c	Number (18, 0)
Input or output with API (with any data size)	minQty	Integer

**Rule**

The rule record associated with the coupon.

Usage	Field Name	Data Type
Input on object	ccrz__Rule__c	Lookup Relationship (ccrz__E_Rule__c)
Input or output with API (with any data size)	rule	ID

**Rule Name**

Formula that generates an escaped HTML string for the coupon code.

Usage	Field Name	Data Type
Input on object	ccrz__RuleName__c	Formula (Text)
Input or output with API (with any data size)	ruleName	String

**Rule Type**

Whether the coupon's discount applies to a cart total amount, shipping amount, or certain products or products with certain spec values. For usage, see [Guidelines When Creating Coupons for Your Storefront](#).

Usage	Field Name	Data Type
Input on object	ccrz__RuleType__c	Picklist
Input or output with API (with any data size)	ruleType	String

**Source Minimum Quantity**

The number of source products that must exist in the cart before the target product discount is valid. For usage, see:

- [Offer a Buy-One, Get-One Discount on Products](#)

Usage	Field Name	Data Type
Input on object	ccrz__SourceMinQty__c	Currency (16, 2)
Input or output with API (with any data size)	sourceMinQty	Decimal

**Source Product**

The product that must exist in the cart before the target product discount is valid. For usage, see:

- [Offer a Buy-One, Get-One Discount on Products](#)

Usage	Field Name	Data Type
Input on object	ccrz__SourceProduct__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	sourceProduct	ID

**Source Relation**

Whether the coupon dependency is a product or spec. For usage, see:

- [Offer a Buy-One, Get-One Discount on Products](#)

Usage	Field Name	Data Type
Input on object	ccrz__SourceRelation__c	Picklist
Input or output with API (with any data size)	sourceRelation	String

**Source Spec**

The required spec definition. For usage, see:

- [Offer a Discount on a Product When Purchasing Any Product with a Specific Spec Value](#)

Usage	Field Name	Data Type
Input on object	ccrz__SourceSpec__c	Lookup Relationship ( <a href="#">ccrz__E_Spec__c</a> )
Input or output with API (with any data size)	sourceSpec	ID

**Source Spec Minimum Quantity**

The number of products with the required spec value that must exist in the cart before the discount is valid. For usage, see:

- [Offer a Discount on a Product When Purchasing Any Product with a Specific Spec Value](#)

Usage	Field Name	Data Type
Input on object	ccrz__SourceSpecMiniQty__c	Number (18, 0)
Input or output with API (with any data size)	sourceSpecMiniQty	Integer

**Source Spec Value**

The required spec value. For usage, see:

- [Offer a Discount on a Product When Purchasing Any Product with a Specific Spec Value](#)

Usage	Field Name	Data Type
Input on object	ccrz__SourceSpecValue__c	Text (100)
Input or output with API (with any data size)	sourceSpecValue	String

**Start Date**

The first day when the coupon is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date

Usage	Field Name	Data Type
Input or output with API (with any data size)	startDate	Date

**Storefront**

The storefront where the coupon is available.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Picklist (Multi-select)
Input or output with API (with any data size)	storefront	String

**Target Product**

The product that you want the discount to apply to. For usage, see:

- [Offer a Buy-One, Get-One Discount on Products](#)
- [Offer a Discount on a Product When Purchasing Any Product with a Specific Spec Value](#)

Usage	Field Name	Data Type
Input on object	ccrz__TargetProduct__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	targetProduct	ID

**Target Quantity**


The number products that can receive the discount during checkout. For usage, see:

- [Offer a Buy-One, Get-One Discount on Products](#)
- [Offer a Discount on a Product When Purchasing Any Product with a Specific Spec Value](#)

Usage	Field Name	Data Type
Input on object	ccrz__TargetQty__c	Number (18, 0)
Input or output with API (with any data size)	targetQty	Integer

**Target Relation**

Whether the object of the discount is a product or product with a particular spec value.

 **Note:** Currently, we recommend that you only select Product for this field.

For usage, see:

- [Offer a Buy-One, Get-One Discount on Products](#)

- [Offer a Discount on a Product When Purchasing Any Product with a Specific Spec Value](#)

Usage	Field Name	Data Type
Input on object	ccrz__TargetRelation__c	Picklist
Input or output with API (with any data size)	targetRelation	String

#### Target Spec

Currently not recommended.

Usage	Field Name	Data Type
Input on object	ccrz__TargetSpec__c	Lookup Relationship ( <a href="#">ccrz__E_Spec__c</a> )
Input or output with API (with any data size)	targetSpec	ID

#### Target Spec Quantity

Currently not recommended.

Usage	Field Name	Data Type
Input on object	ccrz__TargetSpecQty__c	Number (18, 0)
Input or output with API (with any data size)	targetSpecQty	Integer

#### Target Spec Value

Currently not recommended.

Usage	Field Name	Data Type
Input on object	ccrz__TargetSpecValue__c	Text (100)
Input or output with API (with any data size)	targetSpecValue	String

#### Total Used

The number of times the coupon has been applied. This value is updated automatically. For usage, see [Guidelines When Creating Coupons for Your Storefront](#).

Usage	Field Name	Data Type
Input on object	ccrz__TotalUsed__c	Number (18, 0)

Usage	Field Name	Data Type
Input or output with API (with any data size)	totalUsed	Integer

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Coupon Type	ccrz__CouponType__c	Picklist

## Related Lists

A coupon record can include the following related list that describes the coupon's relationship to shopping carts where the coupon is applied. Although this related data isn't defined on the CC Coupon object, a `ccrz.ccApiCoupon.fetch` call can return related object records. This related list is nested under the `ccrz.ccApiProduct.COUPONLIST` return key.

### ECartCouponsS

List<Map<String, Object>> of `ccrz__E_CartCoupon__c`.

Consideration	Value
Default data size returned with	Any
<code>ccrz.ccApi.SZ_REL</code> request value	E_CartCoupons__r

SEE ALSO:

[ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)

[ccrz\\_\\_E\\_Product\\_\\_c](#)

[ccrz\\_\\_E\\_ProductSpec\\_\\_c](#)

[ccrz\\_\\_E\\_Spec\\_\\_c](#)

[Salesforce Help Offer Coupons for Discounts on Storefront Purchases](#)

## ccrz\_\_E\_FeaturedProduct\_\_c

Represents the details for a product shown in the Featured Products or Spotlight Products widget on your storefront's Home page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiCategory](#)

**Fetch sizing:** The [ccrz.ccApiCategory.fetch](#) method returns featured product data when the input data specifies `ccrz.ccApiCategory.FEATURED => 'true'`.

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

### DescriptionRT

Text to show in the Featured Products or Spotlight Products widget. If specified, this text overrides the product's existing short description. To show the product's existing short description, leave this field blank.

Usage	Field Name	Data Type
Input on object	ccrz__DescriptionRT__c	Text Area (Rich) (32768)
Input or output with API (with any data size)	descriptionRT	String

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	enabled	Boolean

### Featured Product ID

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__FeaturedProductId__c	Text (255)
Input or output with API (with any data size)	featuredProductId	String

**Featured Product Name**

An internal-facing name for the record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfdcName	String

**Product Item**

The product that you want to showcase.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Master-Detail Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	product	ID

**Sequence**

The order of the product within the Featured Products or Spotlight Products widget. Lower Sequence values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with any data size)	sequence	Integer

**Storefront**

Storefront where the featured product appears.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Picklist (Multi-select)
Input or output with API (with any data size)	storefront	String

**Type**

Whether the product is included in the Featured Products or Spotlight Products widget.

Usage	Field Name	Data Type
Input on object	ccrz__Type__c	Picklist



Usage	Field Name	Data Type
Input or output with API (with any data size)	type	String

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Promotion Description	ccrz__Description__c	Text Area (255)

SEE ALSO:

[Salesforce Help Show Spotlight Products and Featured Products](#)

## ccrz\_\_E\_FeaturedProductI18N\_\_c

Represents a featured product translated for a locale that your storefront supports.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

The API doesn't return fields on the `ccrz__E_FeaturedProductI18N__c` record directly. When the `ccrz.ccApiCategory.fetch` method's input map includes the `ccrz.ccApiCategory.CATEGORYLOCALE` key and sets `ccrz.ccApiCategory.FEATURED => 'true'`, the method:

1. References `ccrz__E_FeaturedProductI18N__c` records for the specified locale.
2. Returns `ccrz__E_FeaturedProduct__c` records, translated with the localized data.

## Fields

### CC Featured Product

The featured product to localize.

Usage	Field Name	Data Type
Input on object	ccrz__FeaturedProduct__c	Master-Detail Relationship ( <a href="#">ccrz__E_FeaturedProduct__c</a> )
Input or output with API (with any data size)	featuredProduct	ID

### DescriptionRT

The localized description for the featured product, which displays on the storefront and overrides the product's localized short description.

Usage	Field Name	Data Type
Input on object	ccrz__DescriptionRT__c	Text Area (Rich) (32768)
Input or output with API (with any data size)	descriptionRT	String

### FeaturedProductI18N Name

An internal-facing name for the record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfdcName	String

### Locale

The [storefront locale](#) that the featured product is localized for.

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (25)
Input or output with API (with any data size)	locale	String

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Description	ccrz__Description__c	Text Area (Long) (32768)

SEE ALSO:

[Salesforce Help Localize Spotlight Products and Featured Products for a Translated Storefront](#)

## ccrz\_\_E\_Invoice\_\_c

Represents an invoice that B2B Commerce for Visualforce creates for a user after the user places an order.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiInvoice](#)

**Fetch sizing:** To specify which fields the [ccrz.ccApiInvoice.fetch](#) method returns for each invoice record, [request a specific data size](#) for the `ccrz.ccApiInvoice.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Bill Address

The contact address record used for billing.

Usage	Field Name	Data Type
Input on object	ccrz__BillAddress__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_M</b> or larger)	billAddress	ID

**Bill To**

The account used for billing. The Bill To and Sold To account are typically the same.

Usage	Field Name	Data Type
Input on object	ccrz__BillTo__c	Lookup Relationship ( <a href="#">Account</a> )
Input or output with API (with <b>SZ_L</b> or larger)	billTo	ID
	billToR	Map< <a href="#">String</a> , <a href="#">Object</a> > that includes the account name and account group.

**CC Invoice Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**CC Order**

The order that the invoice was created for.

Usage	Field Name	Data Type
Input on object	ccrz__CCOrder__c	Lookup Relationship ( <a href="#">ccrz__E_Order__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	CCOrder	ID

**Comments**

Optional free text for the invoice.

Usage	Field Name	Data Type
Input on object	ccrz__Comments__c	Text Area (Long) (32768)
Input or output with API (with <b>SZ_M</b> or larger)	comments	String

**Contact**

The contact record for populating the invoice record.

Usage	Field Name	Data Type
Input on object	ccrz__Contact__c	Lookup Relationship ( <a href="#">Contact</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Currency Code**

Currency of the order.

Usage	Field Name	Data Type
Input on object	ccrz__CurrencyCode__c	Text (10)
Input or output with API (with <b>SZ_M</b> or larger)	currencyCode	String

**Date Due**

The date when the invoice balance is due.

Usage	Field Name	Data Type
Input on object	ccrz__DateDue__c	Date
Input or output with API (with <b>SZ_M</b> or larger)	dateDue	Date

**Date Issued**

The date the invoice was created.

Usage	Field Name	Data Type
Input on object	ccrz__DateIssued__c	Date
Input or output with API (with <b>SZ_M</b> or larger)	dateIssued	Date

**Invoice ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__InvoiceId__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	invoiceId	String

**Original Amount**

The original total amount billed on the invoice. This value can be positive or negative. Don't change this value after creating in the invoice.

Usage	Field Name	Data Type
Input on object	ccrz__OriginalAmount__c	Currency (16, 2)
Input or output with API (with <b>SZ_M</b> or larger)	originalAmount	Decimal

**Paid Amount**

The difference between the original amount and the remaining amount.

Usage	Field Name	Data Type
Input on object	ccrz__PaidAmount__c	Formula (Currency)
Input or output with API (with <b>SZ_M</b> or larger)	paidAmount	Decimal

**Parent Invoice**

Reference to summary parent invoice.

Usage	Field Name	Data Type
Input on object	ccrz__ParentInvoice__c	Lookup Relationship (ccrz__E_Invoice__c)
Input or output with API (with <b>SZ_M</b> or larger)	parentInvoice	ID

**Remaining Amount**

The amount that hasn't been paid on the invoice.

Usage	Field Name	Data Type
Input on object	ccrz__RemainingAmount__c	Currency (16, 2)
Input or output with API (with <b>SZ_M</b> or larger)	remainingAmount	Decimal

**Sold Address**

The contact address for the account for that is responsible for the invoice.

Usage	Field Name	Data Type
Input on object	ccrz__SoldAddress__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	soldAddress	ID

**Sold To**

The account that is responsible for the invoice. The Bill To and Sold To account are typically the same.

Usage	Field Name	Data Type
Input on object	ccrz__SoldTo__c	Lookup Relationship ( <a href="#">Account</a> )
Input or output with API (with <b>SZ_L</b> or larger)	soldTo	ID
	soldToR	Map< <a href="#">String</a> , <a href="#">Object</a> > that includes the account name and account group.

**Status**

Whether the invoice remains open or is closed.

Usage	Field Name	Data Type
Input on object	ccrz__Status__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	status	String

**Storefront**

The storefront where the order was placed.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	storefront	String

**Terms**

Instructions or requirements for the buyer to pay the balance on the invoice.

Usage	Field Name	Data Type
Input on object	ccrz__Terms__c	Text (255)
Input or output with API (with <b>SZ_XL</b> )	terms	String

**Type**

Whether the invoice is a credit or debit.

Usage	Field Name	Data Type
Input on object	ccrz__Type__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	type	String

**User**

The user record for populating the invoice record.

Usage	Field Name	Data Type
Input on object	ccrz__User__c	Lookup Relationship ( <a href="#">User</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Related Lists**

A record of the `ccrz__E_Invoice__c` object can have multiple related lists that relate the invoice to other types of records, such as invoice items and child invoices. The `ccrz.ccApiInvoice.fetch` method can return data from these related records, depending on the requested data size in the input map.

The reference for each related list indicates which data size the list is returned with by default. You can also use the [ccrz.ccApi.SZ\\_REL flag in your input map](#) to request a particular list, regardless of requested data size:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
    ccrz.ccApi.SZ_REL => New List<String>{'InvoiceItems__r'}
  }
}
```

**childInvoices**

`List<Map<String, Object>>` of `ccrz__E_Invoice__c` records that are children of the current `ccrz__E_Invoice__c`.

Consideration	Value
Default data size returned with	<b>SZ_M</b> or larger
<code>ccrz.ccApi.SZ_REL</code> request value	<code>ChildInvoices__r</code>

**invoiceItems**

`List<Map<String, Object>>` of `ccrz__E_InvoiceItem__c`.



Consideration	Value
Default data size returned with	<b>SZ_M</b> or larger
<code>ccrz.ccApi.SZ_REL</code> request value	InvoiceItems__r

SEE ALSO:

[Salesforce Help Generate an Invoice for a Buyer's Order](#)

## ccrz\_\_E\_InvoiceItem\_\_c

Represents a line item referenced from an invoice, such as a product or a fee.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** `ccrz.ccApiInvoice`

**Fetch sizing:** The `ccrz.ccApiInvoice.fetch` method returns the `invoiceItemsS` related list when you [request at least a medium data size](#) for the `ccrz.ccApiCategory.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'InvoiceItems__r'}
    }
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

### CC Invoice Item Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

### CC Order

The order that the invoice was created for.

Usage	Field Name	Data Type
Input on object	ccrz__CCOrder__c	Lookup Relationship ( <a href="#">ccrz__E_Order__c</a> )
Input or output with API (with any data size)	CCOrder	ID
	CCOrderR	Map<String, Object> that describes the order.

### CC Order Item

Order item that the invoice item corresponds to.

Usage	Field Name	Data Type
Input on object	ccrz__CCOrderItem__c	Lookup Relationship ( <a href="#">E_OrderItem__c</a> )
Input or output with API (with any data size)	CCOrderItem	ID
	CCOrderItemR	Map<String, Object> that describes the order item.

### Comments

Optional free text for the invoice item.

Usage	Field Name	Data Type
Input on object	ccrz__Comments__c	Text Area (Long) (32768)
Input or output with API (with any data size)	comments	String

**Invoice**

The invoice that the items belong to.

Usage	Field Name	Data Type
Input on object	ccrz__Invoice__c	Master-Detail Relationship ( <a href="#">ccrz__E_Invoice__c</a> )
Input or output with API (with any data size)	invoice	ID
	invoiceR	Map<String, Object> that describes the invoice.

**Invoice Item ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__InvoiceItemId__c	Text (255)
Input or output with API (with any data size)	invoiceItemId	String

**Original Amount**

The original amount of the item billed on the invoice. This value can be positive or negative.

 **Note:** Don't change this value after creating the invoice.

Usage	Field Name	Data Type
Input on object	ccrz__OriginalAmount__c	Currency (16, 2)
Input or output with API ((with any data size)	originalAmount	Decimal

**Sequence**

The order of the item among other items for the invoice. Lower Sequence values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (18, 0)
Input or output with API (with any data size)	sequence	Integer

**Status**

Whether the invoice item remains open or is closed. By default, the status of invoice items changes to Closed when the invoice's balance is paid in full.

Usage	Field Name	Data Type
Input on object	ccrz__Status__c	Text (255)
Input or output with API (with any data size)	status	String

**Type**

Type of item, such as detail, comment, or payment.

Usage	Field Name	Data Type
Input on object	ccrz__Type__c	Text (255)
Input or output with API (with any data size)	type	String

SEE ALSO:

[Salesforce Help Generate an Invoice for a Buyer's Order](#)

## ccrz\_\_E\_Menu\_\_c

Represents a top-level entry in the storefront's navigation bar. Refresh the cached menu data whenever you modify menu records so that the storefront reflects the most recent menu data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiMenu](#)

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

## Fields

### Category

A category that the menu links to when Link Type is Category.

Usage	Field Name	Data Type
Input on object	ccrz__Category__c	Lookup Relationship (ccrz__E_Category__c)
Input or output with API (with any data size)	category	ID

### CC Menu Auto Number

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfid	ID

### DisplayName

Text that you want to appear for the menu in the navigation bar.

Usage	Field Name	Data Type
Input on object	ccrz__DisplayName__c	Text (255)
Input or output with API (with any data size)	displayName	String

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

### End Date

The last day that the menu is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Link Type**

The destination for the menu's link. For valid values and use cases, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_menu.htm](https://help.salesforce.com/articleView?id=b2b_commerce_menu.htm).

Usage	Field Name	Data Type
Input on object	ccrz__LinkType__c	Picklist
Input or output with API (with any data size)	mType	String

**MenuId**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__MenuId__c	Text (128)
Input or output with API (with any data size)	menuId	String

**New Window**

Whether a new window or tab opens in the buyer's browser when they click the menu. This field is evaluated only when Link Type is URL.

Usage	Field Name	Data Type
Input on object	ccrz__NewWindow__c	Checkbox
Input or output with API (with any data size)	openInNewWindow	Boolean

**Product**

A product that the menu links to when Link Type is Product.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	product	ID

Usage	Field Name	Data Type
	productR	Map<String, Object>, which specifies other product information, including SKU and ID.

### Sequence

Order of the menu within the navigation bar, relative to other menus assigned to the same storefront. Lower values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (5, 0)
Input or output with API (with any data size)	sequence	Integer

### Show in My Account Menu

Whether the menu is used for linking to sections of the My Account page in the header, including the B2B Commerce Header component in an Experience Builder site. For usage, see

[https://help.salesforce.com/articleView?id=b2b\\_commerce\\_menu\\_my\\_account.htm](https://help.salesforce.com/articleView?id=b2b_commerce_menu_my_account.htm).



**Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	ccrz__ShowInMyAccountMenu__c	Checkbox
Input or output with API (with any data size)	showInMyAccountMenu	Boolean

### Start Date

The first day that the menu is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Store ID

The name of the storefront where the menu appears.

Usage	Field Name	Data Type
Input on object	ccrz__StoreID__c	Text (128)

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**URL**

The URL that the menu links to when Link Type is URL.

Usage	Field Name	Data Type
Input on object	ccrz__URL__c	Text Area (Long) (32768)
Input or output with API (with any data size)	linkURL	ID

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Location Type	ccrz__LocationType__c	Picklist
Page Location	ccrz__PageLocation__c	Picklist
New Window Property	ccrz__NewWindowProperty__c	Text Area (255)

SEE ALSO:

[Salesforce Help Refresh the Menu Cache](#)

**ccrz\_\_E\_Menu18N\_\_c**

Represents an internationalized menu record for a specific locale.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



## API Usage

The API doesn't return fields on the `ccrz__E_MenuI18N__c` record directly. When the `ccrz.ccApiMenu.fetch` method's input map includes the `ccrz.ccApiMenu.LOCALE` key, the method:

1. References `ccrz__E_MenuI18N__c` records for the specified locale.
2. Returns `ccrz__E_Menu__c` records, translated with the localized data.

## Fields

### CC Menu

The menu record to localize.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Menu__c</code>	Master-Detail Relationship ( <code>ccrz__E_Menu__c</code> )
Input or output with API (with any data size)	Not applicable	

### Display Name

The localized text for the menu entry in the navigation bar.

Usage	Field Name	Data Type
Input on object	<code>ccrz__DisplayName__c</code>	Text (255)
Input or output with API (with any data size)	Provides the value for the <code>displayName</code> field when the API queries localized menu records.	

### Language

Code for the locale selected from the Locale picklist on a CC Menu I18N record.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Language__c</code>	Text (25)
Input or output with API (with any data size)	Not applicable	

### Locale

An available locale to show in the Locale picklist on a CC Menu I18N record. This field doesn't store any value on the actual record—the `Language` field specifies the selected value.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Locale__c</code>	Text (25)

Usage	Field Name	Data Type
Input or output with API (with any data size)	Not applicable	

**Menu I18N AutoNumber**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	Not applicable	

**Menu I18N Ext Id**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__MenuI18NId__c	Text (128)
Input or output with API (with any data size)	Not applicable	

SEE ALSO:

[Salesforce Help Localize Menus for an Internationalized Storefront](#)

## ccrz\_\_E\_MenuItem\_\_c

Represents a nested menu entry in a menu in the storefront's navigation bar. Refresh the cached menu data whenever you modify menu item records so that the storefront reflects the most recent menu data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiMenu`

The `ccrz.ccApiMenu.fetch` method returns `ccrz__E_MenuItem__c` record data in the `children` field of the `ccrz.ccApiMenu.MENULIST` return data key.

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

## Fields

### Category

A category that the menu item links to when Link Type is Category.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Category__c</code>	Lookup Relationship ( <code>ccrz__E_Category__c</code> )
Input or output with API (with any data size)	<code>category</code>	ID

### CC Menu Item Auto Number

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	<code>sfid</code>	ID

### DisplayName

Text that you want to appear for the menu item in the menu.

Usage	Field Name	Data Type
Input on object	<code>ccrz__DisplayName__c</code>	Text (255)
Input or output with API (with any data size)	<code>displayName</code>	String

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Enabled__c</code>	Checkbox

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**End Date**

The last day that the menu item is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Link Type**

The destination for the menu item's link. For valid values and use cases, see [Provide Intuitive Storefront Navigation with Menus](#).

Usage	Field Name	Data Type
Input on object	ccrz__LinkType__c	Picklist
Input or output with API (with any data size)	mType	String

**Menu**

The menu that the menu item is nested under. Use this field if the menu item represents the second level of a menu: a parent menu and a menu item nested within the menu.

Usage	Field Name	Data Type
Input on object	ccrz__Menu__c	Lookup Relationship ( <a href="#">ccrz__E_Menu__c</a> )
Input or output with API (with any data size)	menu	ID

**Menu Item Id**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__MenuItemId__c	Text (128)
Input or output with API (with any data size)	menuId	String

**New Window**

Whether a new window or tab opens in the buyer's browser when they click the menu item. This field is evaluated only when Link Type is URL.

Usage	Field Name	Data Type
Input on object	ccrz__NewWindow__c	Checkbox
Input or output with API (with any data size)	openInNewWindow	Boolean

**Parent Menu Item**

The menu item that the current menu item is nested under. Use this field if the menu item represents the third level of a menu: a parent menu and two levels of menu items.

**Note:**

- When this field has a value, leave the `Menu` field blank.
- The managed package doesn't limit how many levels of menu items you create in your org's data. However, by default, the storefront can show at most three levels of a menu: a parent menu and two levels of nested menu items. This limit is based on both of the following default resources included in the managed package:
  - The default Bootstrap library, which allows only two levels per menu
  - A CSS file, `shimbs3.css`, which extends the Bootstrap limit to show three levels per menu

If your org's data specifies more levels of nested menus, those further levels don't display by default. To show more than three levels, you can extend the default `shimbs3.css` and extend the `ccrz.cc_hk_UserInterface` extension point class in the API to reference your subscriber code.

Usage	Field Name	Data Type
Input on object	ccrz__ParentMenuItem__c	Lookup Relationship (ccrz__E_MenuItem__c)
Input or output with API (with any data size)	parentMenuItem	ID

**Product**

A product that the menu item links to when Link Type is Product.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Lookup Relationship (ccrz__E_Product__c)
Input or output with API (with any data size)	product	ID
	productR	Map<String, Object>, which specifies other product information, including SKU and ID.

**Sequence**

Order of the menu item within the menu, relative to other menu items with the same parent menu. Lower values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (5, 0)
Input or output with API (with any data size)	sequence	Integer

**Start Date**

The first day that the menu item is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with any data size)	The API doesn't return data for this field.	

**URL**

The URL that the menu links to when Link Type is URL.

Usage	Field Name	Data Type
Input on object	ccrz__URL__c	Text Area (Long) (32768)
Input or output with API (with any data size)	linkURL	ID

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
New Window Property	ccrz__NewWindowProperty__c	Text Area (255)

SEE ALSO:

[Salesforce Help Refresh the Menu Cache](#)

**ccrz\_\_E\_MenuItem18N\_\_c**

Represents an internationalized menu item record for a specific locale.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

The API doesn't return fields on the `ccrz__E_MenuItemI18N__c` record directly. When the `ccrz.ccApiMenu.fetch` method's input map includes the `ccrz.ccApiMenu.LOCALE` key, the method:

1. References `ccrz__E_MenuItemI18N__c` records for the specified locale.
2. Returns `ccrz__E_MenuItem__c` records in the `children` field of the `ccrz.ccApiMenu.MENULIST` return data key. The menu items are translated with the localized data.

## Fields

### Display Name

The localized text for the menu item.

Usage	Field Name	Data Type
Input on object	<code>ccrz__DisplayName__c</code>	Text (255)
Input or output with API (with any data size)	Provides the value for the <code>displayName</code> field when the API queries localized menu records.	

### Language

Code for the locale selected from the Locale picklist on a CC Menu Item I18N record.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Language__c</code>	Text (25)
Input or output with API (with any data size)	Not applicable	

### Locale

An available locale to show in the Locale picklist on a CC Menu I18N record. This field doesn't store any value on the actual record—the `Language` field specifies the selected value.

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (25)
Input or output with API (with any data size)	Not applicable	

**Menu Item**

The menu item record to localize.

Usage	Field Name	Data Type
Input on object	ccrz__MenuItem__c	Master-Detail Relationship (ccrz__E_MenuItem__c)
Input or output with API (with any data size)	Not applicable	

**Menu Item I18N AutoNumber**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	Not applicable	

**Menu Item I18N Ext Id**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__MenuItemI18NId__c	Text (128)
Input or output with API (with any data size)	Not applicable	

SEE ALSO:

[Salesforce Help Localize Menus for an Internationalized Storefront](#)

**ccrz\_\_E\_Order\_\_c**

Represents a buyer's completed order, based on a shopping cart that the buyer checked out with.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiOrder](#)

**Fetch sizing:** To specify which fields the [ccrz.ccApiOrder.fetch](#) method returns for each order record, [request a specific data size](#) for the `ccrz.ccApiOrder.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Account

The account associated with the user assigned in the `User` field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Account__c</code>	Lookup Relationship ( <a href="#">Account</a> )
Input or output with API (with <b>SZ_L</b> or larger)	<code>account</code>	ID

### Adjustment Amount

Amount to adjust the order price by after the order is placed. This amount is specified on the cart, and passed to the corresponding `ccrz__E_Order__c` record. You can use your own back-end processing to apply this adjustment without changing the values of other fields.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AdjustmentAmount__c</code>	Currency (16, 2)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_L</b> or larger)	adjustmentAmount	Decimal

**Adjustment Reason**

Text that explains or justifies an adjustment amount. This text is specified on the cart, and passed to the corresponding `ccrz__E_Order__c` record.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AdjustmentReason__c</code>	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	adjustmentReason	String

**Bill To**

The billing address associated with the order. This record is a unique copy of the billing address specified during checkout.

Usage	Field Name	Data Type
Input on object	<code>ccrz__BillTo__c</code>	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with <b>SZ_XL</b> )	billTo	ID

**Cart Restored**


Whether the cart that corresponds to an order that failed to place asynchronously has been reset to Open and made active. You can restore an order only one time. For more information, see [ccrz.cclLogicOrderRestore](#).

 **Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	<code>ccrz__IsCartRestored__c</code>	Checkbox
Input or output with API (with <b>SZ_M</b> or larger)	isCartRestored	Boolean

**CCEmailAddress**

Whitespace-separated list of email addresses for the CC line of the order confirmation email. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_page\\_checkout\\_cc\\_email.htm](https://help.salesforce.com/articleView?id=b2b_commerce_page_checkout_cc_email.htm).

 **Tip:** We recommend that you populate this field only for orders that contain a low volume of cart items.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CCEmailAddress__c</code>	Text Area (Long) (32768)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_XL</b> )	CCEmailAddress	String

**CCSubscription**

A subscription associated with the order.

Usage	Field Name	Data Type
Input on object	ccrz__CCSubscription__c	Lookup Relationship ( <a href="#">ccrz__E_Subscription__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	CCSubscription	ID

**Contact**

The contact associated with the user assigned in the User field.

Usage	Field Name	Data Type
Input on object	ccrz__Contact__c	Lookup Relationship ( <a href="#">Contact</a> )
Input or output with API (with <b>SZ_M</b> or larger)	contact	ID

**Currency ISO Code**

The ISO 4217 currency code for the order, such as *USD* or *JPY*.

Usage	Field Name	Data Type
Input on object	ccrz__CurrencyISOCODE__c	Text (10)
Input or output with API (with <b>SZ_L</b> or larger)	currencyISOCODE	String

**Effective Account ID**

The effective account associated with the order. For usage, see [Manage Carts for Effective Accounts](#).

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveAccountID__c	Text (20)
Input or output with API (with <b>SZ_S</b> or larger)	effectiveAccountID	String

**Email**

Buyer's email address.

Usage	Field Name	Data Type
Input on object	ccrz__BuyerEmail__c	Email
Input or output with API (with <b>SZ_XL</b> )	buyerEmail	String

**Encrypted ID**

Semi-random UUID to refer to the order in page URLs.



**Note:** In an encrypted ID, avoid XML characters—such as <, >, or & characters—which are escaped by the default Visualforce remoting request for fetching order history.

Usage	Field Name	Data Type
Input on object	ccrz__EncryptedId__c	Text (250)
Input or output with API (with <b>SZ_S</b> or larger)	encryptedId	String

**First Name**

Buyer's first name.

Usage	Field Name	Data Type
Input on object	ccrz__BuyerFirstName__c	Text (40)
Input or output with API (with <b>SZ_XL</b> )	buyerFirstName	String

**Is Anonymous**

Whether a guest buyer placed the order.

Usage	Field Name	Data Type
Input on object	ccrz__IsAnonymous__c	Checkbox
Input or output with API (with <b>SZ_L</b> or larger)	isAnonymous	Boolean

**Last Name**

Buyer's last name.

Usage	Field Name	Data Type
Input on object	ccrz__BuyerLastName__c	Text (80)
Input or output with API (with <b>SZ_XL</b> )	buyerLastName	String

**Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Note**

A shipping note entered during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__Note__c	Text Area (Long) (1000)
Input or output with API (with <b>SZ_XL</b> )	note	String

**Number of All Order Items**

The total number of items in the order, including both major and minor line items.



**Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	ccrz__NumberOfAllOrderItems__c	Roll-Up Summary (COUNT <a href="#">ccrz__E_OrderItem__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	numberOfAllOrderItems	Integer

**Number of Major Order Items**

The total number of major line items in the order.



**Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	ccrz__NumberOfMajorOrderItems__c	Roll-Up Summary (COUNT <a href="#">ccrz__E_OrderItem__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	numberOfMajorOrderItems	Integer

**Order**

A parent order that for the current order. This field is used when fulfilling subscriptions:

- The parent order describes the upfront order that placed the subscription.

- The child order describes a recurring order for fulfilling the subscription.

Usage	Field Name	Data Type
Input on object	ccrz__Order__c	Lookup Relationship (ccrz__E_Order__c)
Input or output with API (with <b>SZ_M</b> or larger)	order	ID

**Order Date**

The date when the buyer placed the order.

Usage	Field Name	Data Type
Input on object	ccrz__OrderDate__c	Date
Input or output with API (with <b>SZ_M</b> or larger)	orderDate	Date

**Order ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__OrderId__c	Text (128)
Input or output with API (with <b>SZ_M</b> or larger)	orderId	String

**Order Name**

The `ccrz__CartName__c` value copied from the corresponding `ccrz__E_Cart__c` record.

Usage	Field Name	Data Type
Input on object	ccrz__Name__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	name	String

**Order Name (Duplicated)**

A copy of the order's `Name` auto-number value, for use in the order confirmation email template.

Usage	Field Name	Data Type
Input on object	ccrz__OrderName__c	Formula (Text)
Input or output with API (with <b>SZ_M</b> or larger)	orderName	String

**Order Number**

The digit portion of the order's Name auto-number value, plus 1,000,000. For example, for an order with an auto-numbered Name of O-000000018, the Order Number is 1,000,018.

Usage	Field Name	Data Type
Input on object	ccrz__OrderNumber__c	Number (18, 0)
Input or output with API (with <b>SZ_M</b> or larger)	orderNumber	Integer

**Order Seller**

A seller associated with the order.

Usage	Field Name	Data Type
Input on object	ccrz__OrderSeller__c	Lookup Relationship ( <a href="#">ccrz__E_Seller__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Order Status**

Status of the order shown in the My Orders section of the My Account page. By default, B2B Commerce for Visualforce sets the following statuses:

**Error**

The order failed to place.

**In Creation**

The `ccrz.ccApiCart.placeAsync` method is placing the order asynchronously.

**On Hold**

The order indicates a tax exemption during checkout.

**Order Submitted**

An order created after checkout completes.



**Note:** The Error and In Creation values are available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

The managed package also defines the following picklist values, but doesn't define any default behavior that sets or changes these values. To set other status values, extend the appropriate class in the B2B Commerce for Visualforce API.

Bill of Material	In Process	Returned
Cancel Submitted	Partial Shipped	Shipped
Cancelled	Open	
Completed	Return Submitted	

Usage	Field Name	Data Type
Input on object	ccrz__OrderStatus__c	Picklist
Input or output with API (with <b>SZ_M</b> or larger)	orderStatus	String

**Originated Cart**

The cart used for placing the order.

Usage	Field Name	Data Type
Input on object	ccrz__OriginatedCart__c	Lookup Relationship ( <a href="#">ccrz__E_Cart__c</a> on page 25)
Input or output with API (with <b>SZ_M</b> or larger)	originatedCart	ID

**Phone**

Buyer's phone number.

Usage	Field Name	Data Type
Input on object	ccrz__BuyerPhone__c	Phone
Input or output with API (with <b>SZ_XL</b> )	buyerPhone	String

**Request Date**

The delivery date that the buyer requested during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__RequestDate__c	Date
Input or output with API (with <b>SZ_L</b> or larger)	requestDate	Date

**Ship Amount**

Amount calculated for shipping the order during checkout. This value is calculated based on shipping rates and product quantities in the cart, and passed to the corresponding `ccrz__E_Order__c` record. [Specify Shipping Rates for a Product.](#)

**Ship Complete**

Whether the buyer selected the Ship Complete option during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__ShipComplete__c	Checkbox



Usage	Field Name	Data Type
Input or output with API (with <b>SZ_M</b> or larger)	shipComplete	Boolean

### Ship Discount Amount

Amount to adjust the shipping cost by after the order is placed. This amount is specified on the cart, and passed to the corresponding `ccrz__E_Order__c` record. You can use your own back-end processing to apply this adjustment without changing the values of other fields.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ShipDiscountAmount__c</code>	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	shipDiscountAmount	Decimal

### Ship Method

The shipping method selected during checkout.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ShipMethod__c</code>	Text (50)
Input or output with API (with <b>SZ_L</b> or larger)	shipMethod	String

### Ship Structure

The checkout experience used for grouping the cart's items for shipment. If your storefront uses line-level independence (LLI), this value is `LLI`. Otherwise, this value is empty.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ShipStructure__c</code>	Text (128)
Input or output with API (with <b>SZ_S</b> or larger)	shipStructure	String

### Ship To

The shipping address associated with the cart. This record is a unique copy of the shipping address specified during checkout.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ShipTo__c</code>	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	shipTo	ID

**Skip Tax Calculation**

Whether tax was calculated during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__SkipTax__c	Checkbox
Input or output with API (with <b>SZ_L</b> or larger)	skipTax	Boolean

**Split Parent Order**

The source order that the current order was split from. For more information, see:

- [ccrz.ccApiOrder.getOrdersToSplit](#)
- [ccrz.ccApiOrder.splitOrder](#)
- [ccrz.ccSplitOrderJob](#)

Usage	Field Name	Data Type
Input on object	ccrz__SplitParentOrder__c	Lookup Relationship (ccrz__E_Order__c)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Split Processed**

Whether the order has been evaluated for splitting. When this field is *TRUE*, processes that split orders ignore this order because it's already been evaluated. For more information, see:

- [ccrz.ccApiOrder.getOrdersToSplit](#)
- [ccrz.ccApiOrder.splitOrder](#)
- [ccrz.ccSplitOrderJob](#)

Usage	Field Name	Data Type
Input on object	ccrz__SplitProcessed__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Storefront**

The storefront where the order was placed.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	storefront	String

**Subtotal Amount**

The total entitled price of all items in the order.

Usage	Field Name	Data Type
Input on object	ccrz__SubtotalAmount__c	Roll-Up Summary (SUM <a href="#">ccrz__E_OrderItem__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	subtotalAmount	Decimal

**Tax Amount**

Tax amount calculated during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__TaxAmount__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	taxAmount	Decimal

**Tax Exempt Flag**

Whether the order is exempt from tax. This value is specified on the cart, and passed to the corresponding `ccrz__E_Order__c` record.

Usage	Field Name	Data Type
Input on object	ccrz__TaxExemptFlag__c	Checkbox
Input or output with API (with <b>SZ_L</b> or larger)	taxExemptFlag	Boolean

**Tax Subtotal Amount**

The total taxable price of all items in the order.

Usage	Field Name	Data Type
Input on object	ccrz__TaxSubTotalAmount__c	Roll-Up Summary (SUM <a href="#">ccrz__E_OrderItem__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	taxSubTotalAmount	Decimal

**Total Amount**

Total cost calculated for the order, including:

- Ship amount
- Ship discount amount

- Subtotal amount
- Tax amount
- Adjustment amount
- Total surcharge

Usage	Field Name	Data Type
Input on object	ccrz__TotalAmount__c	Formula (Currency)
Input or output with API (with <b>SZ_L</b> or larger)	totalAmount	Decimal

### Total Discount

Total amount discounted from the order.

Usage	Field Name	Data Type
Input on object	ccrz__TotalDiscount__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	totalDiscount	Decimal

### Total Surcharge

Charges other than taxes or shipping applied to the order.

Usage	Field Name	Data Type
Input on object	ccrz__TotalSurcharge__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	totalSurcharge	Decimal

### User

The user that created the cart used for placing the order.

Usage	Field Name	Data Type
Input on object	ccrz__User__c	Lookup Relationship ( <a href="#">User</a> )
Input or output with API (with <b>SZ_XL</b> )	user	ID

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Company Name	ccrz__BuyerCompanyName__c	Text (80)
Contract ID	ccrz__ContractId__c	Text (50)
Ext Carrier	ccrz__ExtCarrier__c	Text (50)
Ext ShipTo	ccrz__ExtShipTo__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Ext ShipTrack No	ccrz__ExtShipTrackNo__c	Text (250)
Mobile Phone	ccrz__BuyerMobilePhone__c	Phone
Order Source	ccrz__OrderSource__c	Text (255)
Payment Method	ccrz__PaymentMethod__c	Text (50)
PO Number	ccrz__PONumber__c	Text (30)
VerifyExemptionFlag	ccrz__VerifyExemptionFlag__c	Checkbox

## Related Lists

A record of the `ccrz__E_Order__c` object can have multiple related lists that relate the order to other types of records, such as order items. The `ccrz.ccApiOrder.fetch` method can return data from these related records, depending on the requested data size in the input map.

The reference for each related list indicates which data size the list is returned with by default. You can also use the `ccrz.ccApi.SZ_REL` flag in your input map to request a particular list, regardless of requested data size:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XS,
        ccrz.ccApi.SZ_REL => New List<String>{'E_OrderItems__r'}
    }
}
```

### EOrderItemsS

List<Map<String, Object>> of `ccrz__E_OrderItem__c`.

Consideration	Value
Default data size returned with	<b>SZ_M</b> or larger
<code>ccrz.ccApi.SZ_REL</code> request value	<code>E_OrderItems__r</code>

### EOrderItemsGroupsS

List<Map<String, Object>> of `ccrz__E_OrderItemGroup__c`.

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger

Consideration	Value
ccrz.ccApi.SZ_REL request value	E_OrderItemGroups__r

**EOrderShipmentsS**

List<Map<String, Object>> of [ccrz\\_\\_E\\_OrderShipment\\_\\_c](#).

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger
ccrz.ccApi.SZ_REL request value	E_OrderShipments__r

**orderTermsS**

List<Map<String, Object>> of [ccrz\\_\\_E\\_OrderTerm\\_\\_c](#).

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger
ccrz.ccApi.SZ_REL request value	OrderTerms__r



**Note:** For legacy solutions where the New Order configuration of the Checkout (`co.neworder`) module is `FALSE`, the `ccrz.ccApiOrder.fetch` method also returns the `EPaymentDetailsS` related list.

## SEE ALSO:

[Salesforce Help Fulfill a Buyer's Order](#)

[Salesforce Help Enable Asynchronous Order Placement for Checkout](#)

[Salesforce Help Set Product Prices in the Cart from an External Source](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

**ccrz\_\_E\_OrderItem\_\_c**

Represents an item in a buyer's completed order, including major and minor line items or applied coupons.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

Class used with: [ccrz.ccApiOrder](#)

Fetch sizing: To specify which fields the [ccrz.ccApiOrder.fetch](#) or [ccrz.ccApiOrder.fetchOrderItems](#) method returns for each order item record, [request a specific data size](#) for the `ccrz.ccApiOrder.OI_ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.OI_ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Absolute Discount

A fixed discount amount for the item in the order's currency.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AbsoluteDiscount__c</code>	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	<code>absoluteDiscount</code>	Decimal

### Adjustment Amount

Amount to adjust the item's price in the order by. Typically, a customer service representative (CSR) performs this adjustment on behalf of a buyer.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AdjustmentAmount__c</code>	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	<code>adjustmentAmount</code>	Decimal

### Availability Message

Inventory availability information for the product. This value is copied from the `ccrz__AvailabilityMessageRT__c` field on the corresponding [ccrz\\_\\_E\\_ProductInventoryItem\\_\\_c](#) record.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AvailabilityMessage__c</code>	Text Area (255)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_XL</b> )	availabilityMessage	String

**Category**

A category that the product is assigned to. This field is set when a buyer adds the product to the cart from the category's Product List page.

Usage	Field Name	Data Type
Input on object	ccrz__Category__c	Lookup Relationship ( <a href="#">ccrz__E_Category__c</a> )
Input or output with API (with <b>SZ_XL</b> )	category	ID

**Comments**

Text that a buyer added to the order item on the Shopping Cart page.

Usage	Field Name	Data Type
Input on object	ccrz__Comments__c	Text Area (255)
Input or output with API (with <b>SZ_XL</b> )	comments	String

**Coupon**

The coupon record that corresponds to a coupon order item.

Usage	Field Name	Data Type
Input on object	ccrz__Coupon__c	Lookup Relationship ( <a href="#">ccrz__E_Coupon__c</a> )
Input or output with API (with <b>SZ_XL</b> )	coupon	ID

**Display Product**

The product record for an aggregated product. This field is set when the order item is a component product in the aggregated product.

Usage	Field Name	Data Type
Input on object	ccrz__DisplayProduct__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	displayProduct	ID



**External Name**

The name for an externally priced product. This name displays for the order item in order information on the storefront instead of the referenced product's name.

Usage	Field Name	Data Type
Input on object	ccrz__ExtName__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	extName	String

**External SKU**

The SKU for an externally priced product. This SKU displays for the order item in order information on the storefront instead of the referenced product's SKU.

Usage	Field Name	Data Type
Input on object	ccrz__ExtSKU__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	extSKU	String

**Is Subscription**

Whether the order is a subscription.

Usage	Field Name	Data Type
Input on object	ccrz__Is_Subscription__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	isSubscription	Boolean

**Item Total**

Total cost calculated for the order item, including:

- Subtotal amount
- Adjustment amount

Usage	Field Name	Data Type
Input on object	ccrz__ItemTotal__c	Formula (Currency)
Input or output with API (with <b>SZ_XL</b> )	itemTotal	Decimal

**Next Subscription Date**

The date for the next recurring order for a subscription.

Usage	Field Name	Data Type
Input on object	ccrz__Next_Subscription_Date__c	Date

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_XL</b> )	nextSubscriptionDate	Date

**Order**

The order that the items belong to.

Usage	Field Name	Data Type
Input on object	ccrz__OrderItemGroup__c	Master-Detail Relationship ( <a href="#">ccrz__E_Order__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	order	ID
Input or output with API (with <b>SZ_XL</b> )	orderR	Map<String, Object>, which contains a <code>sfdcName</code> field that references the order's auto number.

**Order Item Group**

The shipping group that the item belongs to. This value is set only when your storefront uses line-level independence (LLI) for checkout.

Usage	Field Name	Data Type
Input on object	ccrz__OrderItemGroup__c	Lookup Relationship ( <a href="#">ccrz__E_OrderItemGroup__c</a> )
Input or output with API (with <b>SZ_XL</b> )	orderItemGroup	ID

**Order Item ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__OrderItemId__c	Text (128)
Input or output with API (with <b>SZ_XL</b> )	orderItemId	ID

**Order Item Status**

Status of the order item. By default, B2B Commerce for Visualforce sets the status for an order item to `Order Submitted`. The managed package also defines the following picklist values, but doesn't define any default behavior that sets or changes these values. To set other status values, extend the appropriate class in the B2B Commerce for Visualforce API.

Usage	Field Name	Data Type
Input on object	ccrz__OrderItemStatus__c	Picklist
Input or output with API (with <b>SZ_XL</b> )	orderItemStatus	String

**Order Line Type**

Whether the order item is a:

- Major line item, such as a standard product or a composite product
- Minor line item, such as a component product of a composite product
- Coupon

Usage	Field Name	Data Type
Input on object	ccrz__OrderLineType__c	Picklist
Input or output with API (with <b>SZ_XL</b> )	orderLineType	String

**OrderItemAutoNumber**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Original Item Price**

The buyer's entitled price for the order item before any adjustments are applied.

Usage	Field Name	Data Type
Input on object	ccrz__OriginalItemPrice__c	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	originalItemPrice	Decimal

**Original Quantity**

The original quantity per unit for component products included in kits, bundles, or dynamic kits.

Usage	Field Name	Data Type
Input on object	ccrz__OriginalQuantity__c	Number (16, 2)
Input or output with API (with <b>SZ_XL</b> )	originalQuantity	Integer

**Parent Order Item**

The composite product that represents a kit, bundle, or dynamic kit for which the current order item is a component product.

Usage	Field Name	Data Type
Input on object	ccrz__ParentOrderItem__c	Lookup Relationship (ccrz__E_OrderItem__c)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_L</b> or larger)	parentOrderItem	ID

**Parent Product**

The composite product that represents a kit, bundle, or dynamic kit for which the current order item is a component product.

Usage	Field Name	Data Type
Input on object	ccrz__ParentProduct__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_XL</b> )	parentProduct	ID

**Percent Discount**

The percentage discount for this order item when a coupon is applied.

Usage	Field Name	Data Type
Input on object	ccrz__PercentDiscount__c	Percent (3, 4)
Input or output with API (with <b>SZ_XL</b> )	percentDiscount	Decimal

**Price**

The buyer's entitled price for the order item.

Usage	Field Name	Data Type
Input on object	ccrz__Price__c	Currency (16, 2)
Input or output with API (with <b>SZ_M</b> or larger)	price	Decimal

**Pricing Modifier Types**

Comma-delimited list of pricing modifiers applied to the order item. The only value that the managed package supports by default is *tiered*. For more information about tiered pricing, see [Adjust Product Prices Based on Tiered Quantities](#).



**Tip:** Avoid using the value *tiered* in any customizations or extensions in your subscriber code.

Usage	Field Name	Data Type
Input on object	ccrz__PricingModifierTypes__c	Text (255)
Input or output with API (with <b>SZ_XL</b> )	pricingModifierTypes	String

**Pricing Type**

An internal value that indicates the order item's pricing source.

- When this value is *ATTRGROUP*, B2B Commerce for Visualforce evaluated the order item as part of an attribute-based product.
- When this value is *external*, B2B Commerce for Visualforce evaluated the order item as externally priced.

Usage	Field Name	Data Type
Input on object	ccrz__PricingType__c	Text (255)
Input or output with API (with <b>SZ_XL</b> )	pricingType	String

**Primary Attribute**

The primary attribute selected on the order item's corresponding attribute-based product.

Usage	Field Name	Data Type
Input on object	ccrz__PrimaryAttr__c	Lookup Relationship ( <a href="#">ccrz__E_Attribute__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	primaryAttr	ID
	primaryAttrR	Map<String, Object>, which contains a <code>parentAttribute</code> field that references the primary attribute's parent attribute.

**Product**

The product that corresponds to the order item.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	product	ID
Input or output with API (with <b>SZ_XL</b> )	productR	Map<String, Object>, which contains the following fields: <ul style="list-style-type: none"> <li>• <code>endDate</code></li> <li>• <code>productStatus</code></li> <li>• <code>productType</code></li> <li>• <code>SKU</code></li> <li>• <code>startDate</code></li> </ul>

**Product Name**

The value of the `Name` field on the order item's corresponding [ccrz\\_\\_E\\_Product\\_\\_c](#).

Usage	Field Name	Data Type
Input on object	ccrz__Product_Name__c	Formula (Text)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**ProductType**

The type of product that corresponds to the cart item. For supported values, see [Types of Products for B2B Commerce for Visualforce](#).

Usage	Field Name	Data Type
Input on object	ccrz__ProductType__c	Text (20)
Input or output with API (with <b>SZ_M</b> or larger)	productType	String

**Quantity**

The quantity for the item in the order.

Usage	Field Name	Data Type
Input on object	ccrz__Quantity__c	Number (18, 0)
Input or output with API (with <b>SZ_S</b> or larger)	quantity	Integer

**Recurring Price**

The recurring price amount for the order item's subscription.

Usage	Field Name	Data Type
Input on object	ccrz__RecurringPrice__c	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	recurringPrice	Decimal

**Recurring Price Sub Amount**


Recurring price subtotal for the total quantity of the order item.

Usage	Field Name	Data Type
Input on object	ccrz__RecurringPriceSubAmt__c	Currency (16, 2)
Input or output with API (with <b>SZ_XL</b> )	recurringPriceSubAmt	Decimal

**Request Date**

The date when the buyer added the product to their shopping cart.

Usage	Field Name	Data Type
Input on object	ccrz__RequestDate__c	Date
Input or output with API (with <b>SZ_XL</b> )	requestDate	Date

 **Note:** By default, this field isn't synchronized with the `ccrz__RequestDate__c` field on the parent `ccrz__E_Order__c` object. To implement this synchronization, extend the `ccrz.ccLogicCartAddTo.setCartItemFields` logic service provider method. In your extension, set the `ccrz__RequestDate__c` value on the `ccrz__E_CartItem__c` record to `null`, which causes the `ccrz__E_CartItem__c` record to inherit the `ccrz__RequestDate__c` value from the parent `ccrz__E_Cart__c` record. This value propagates to the corresponding `ccrz__E_OrderItem__c` when the buyer places an order from the cart.

```
global with sharing class customLogicCartAddTo extends ccrz.ccLogicCartAddTo {
    global override Map<String, Object> setCartItemFields(Map<String, Object>
inputData);
        ccrz__E_CartItem__c cartItem = (ccrz__E_CartItem__c)
inputData;
    }
}
```

### Secondary Attribute

The secondary attribute selected on the order item's corresponding attribute-based product.

Usage	Field Name	Data Type
Input on object	ccrz__SecondAttr__c	Lookup Relationship ( <code>ccrz__E_Attribute__c</code> )
Input or output with API (with <b>SZ_S</b> or larger)	secondaryAttr	ID
	secondaryAttrR	Map<String, Object>, which contains a <code>parentAttribute</code> field that references the secondary attribute's parent attribute.

### Selected Attributes

JSON content that describes the selected attributes for the order item's corresponding product.

Usage	Field Name	Data Type
Input on object	ccrz__SelectedAttr__c	Text Area (Long) (1000)
Input or output with API (with <b>SZ_XL</b> )	selectedAttr	ID

**Seller**

The seller that the buyer is purchasing the product from.

Usage	Field Name	Data Type
Input on object	ccrz__Seller__c	Lookup Relationship ( <a href="#">ccrz__E_Seller__c</a> )
Input or output with API (with <b>SZ_XL</b> )	seller	ID
	sellerR	Map<String, Object>, which contains a ID field that references the seller ID.

**Split Source**

The original order item that the current order item was split from. For more information, see [ccrz.ccLogicOrderSplitOrder](#).

Usage	Field Name	Data Type
Input on object	ccrz__SplitSource__c	Lookup Relationship ( <a href="#">ccrz__E_OrderItem__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Store ID**

The storefront where the order's cart was created.

Usage	Field Name	Data Type
Input on object	ccrz__StoreId__c	Text (128)
Input or output with API (with <b>SZ_M</b> or larger)	storeId	String

**Sub Amount**

The entitled subtotal price for the total quantity of the order item.

Usage	Field Name	Data Type
Input on object	ccrz__SubAmount__c	Currency (16, 2)
Input or output with API (with <b>SZ_L</b> or larger)	subAmount	Decimal

**SubProdTerm**

The subscription that corresponds to the order item.



Usage	Field Name	Data Type
Input on object	ccrz__SubProdTerm__c	Lookup Relationship ( <a href="#">ccrz__E_SubProdTerm__c</a> )
Input or output with API (with <b>SZ_XL</b> )	subProdTerm	ID

### Tertiary Attribute

The tertiary attribute selected on the order item's corresponding attribute-based product.

Usage	Field Name	Data Type
Input on object	ccrz__TertiaryAttr__c	Lookup Relationship ( <a href="#">ccrz__E_Attribute__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	tertiaryAttr	ID
	tertiaryAttrR	Map<String, Object>, which contains a parentAttribute field that references the tertiary attribute's parent attribute.

### Unit Of Measure

The unit of measure from the cart item's corresponding product.

Usage	Field Name	Data Type
Input on object	ccrz__UnitOfMeasure__c	Text (32)
Input or output with API (with <b>SZ_L</b> or larger)	unitOfMeasure	String

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Item Label	ccrz__ItemLabel__c	Text (80)
Item Status	ccrz__ItemStatus__c	Picklist
Subscription Auto Renewal	ccrz__Subscription_Auto_Renewal__c	Checkbox
Subscription Duration	ccrz__Subscription_Duration__c	Number (18, 0)
Subscription End Date	ccrz__Subscription_End_Date__c	Date
Subscription Frequency	ccrz__Subscription_Frequency__c	Number (4, 0)
Subscription Start Date	ccrz__Subscription_Start_Date__c	Date

Field Label	Field Name	Data Type
Subscription Status	ccrz__Subscription_Status__c	Picklist
SubscriptionId	ccrz__SubscriptionId__c	Formula (Text)

## SEE ALSO:

[Salesforce Help Fulfill a Buyer's Order](#)

[Salesforce Help Enable Asynchronous Order Placement for Checkout](#)

[Salesforce Help Set Product Prices in the Cart from an External Source](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz\_\_E\_OrderItemGroup\_\_c

Represents a shipping group of order items for line-level independence (LLI) checkout.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** `ccrz.ccApiOrder`

**Fetch sizing:** The `ccrz.ccApiOrder.fetch` method returns the `EOrderItemGroupsS` related list when you [request at least a large data size](#) for the `ccrz.ccApiOrder.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'E_OrderItemGroups__r'}
    }
}
```

```

    }
}

```

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

### CC Order

The order where the shipping group applies.

Usage	Field Name	Data Type
Input on object	ccrz__Order__c	Master-Detail Relationship ( <a href="#">ccrz__E_Order__c</a> )
Input or output with API (with any data size)	order	ID

### Group Name

The shipping group name, used for organizing groups on the Checkout page.

Usage	Field Name	Data Type
Input on object	ccrz__GroupName__c	Text (255)
Input or output with API (with any data size)	groupName	String

### Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Note

Shipping notes for the group.

Usage	Field Name	Data Type
Input on object	ccrz__Note__c	Text Area (Long) (1,000)
Input or output with API (with any data size)	note	String

**OrderItemId**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__OrderItemId__c	Text (128)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Request Date**

The buyer's requested delivery date for the shipping group.

Usage	Field Name	Data Type
Input on object	ccrz__RequestDate__c	Date
Input or output with API (with any data size)	requestDate	Date

**Ship Amount**

The shipping cost calculated for the group, based on shipping rates and product quantities in the group. This value is passed to the corresponding [ccrz\\_\\_E\\_Cart\\_\\_c](#) and [ccrz\\_\\_E\\_Order\\_\\_c](#) records. For more information, see [Specify Shipping Rates for a Product](#).

Usage	Field Name	Data Type
Input on object	ccrz__ShipAmount__c	Currency (16, 2)
Input or output with API (with any data size)	shipAmount	Decimal

**Ship Method**

The shipping method for the group.

Usage	Field Name	Data Type
Input on object	ccrz__ShipMethod__c	Text (50)
Input or output with API (with any data size)	shipMethod	String

**Ship To**

The shipping address for the group.

Usage	Field Name	Data Type
Input on object	ccrz__ShipTo__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with any data size)	shipTo	ID

## ccrz\_\_E\_OrderShipment\_\_c

Represents shipment details for an order.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** [ccrz.ccApiOrder](#)

**Fetch sizing:** The [ccrz.ccApiOrder.fetch](#) method returns the `EOrderShipmentsS` and `EOrderShipmentItemsS` related lists when you [request at least a large data size](#) for the `ccrz.ccApiOrder.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'E_OrderShipments__r'}
    }
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

### Carrier

The shipping service provider.

Usage	Field Name	Data Type
Input on object	ccrz__Carrier__c	Picklist
Input or output with API (with any data size)	carrier	String

### Delivery Date

The expected date when the carrier delivers the shipment.

Usage	Field Name	Data Type
Input on object	ccrz__DeliveryDate__c	Date
Input or output with API (with any data size)	deliveryDate	Date

### Order

The order that the shipment applies to.

Usage	Field Name	Data Type
Input on object	ccrz__Order__c	Master-Detail Relationship ( <a href="#">ccrz__E_Order__c</a> )
Input or output with API (with any data size)	order	ID

### Order Shipment ID

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__OrderShipmentId__c	Text (128)
Input or output with API (with any data size)	orderShipmentId	String

### Order Shipment Number

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number

Usage	Field Name	Data Type
Input or output with API (with any data size)	sfdcName	String

**Shipment Date**

The date when the carrier sends the shipment.

Usage	Field Name	Data Type
Input on object	ccrz__ShipmentDate__c	Date
Input or output with API (with any data size)	shipmentDate	Date

**Shipping Method**

The shipping method selected during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__ShippingMethod__c	Text (255)
Input or output with API (with any data size)	shippingMethod	String

**Tracking Number**

A unique ID for tracking the shipment. To fulfill the tracking, pass this value to the shipping service provider.

Usage	Field Name	Data Type
Input on object	ccrz__TrackingNo__c	Text (255)
Input or output with API (with any data size)	trackingNo	String

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Bill of Lading	ccrz__BillofLading__c	Text (255)

**Related Lists**

The `ccrz.ccApiOrder.fetch` method returns the following related list for each `ccrz__E_OrderShipment__c` record:

**EOrderShipmentItemsS**

List<Map<String, Object>> of [ccrz\\_\\_E\\_OrderShipmentItem\\_\\_c](#).

## ccrz\_\_E\_OrderShipmentItem\_\_c

Represents shipment details for an item in a buyer's completed order.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiOrder](#)

**Fetch sizing:** The [ccrz.ccApiOrder.fetch](#) method returns the [EOrderShipmentsS](#) and [EOrderShipmentItemsS](#) related lists when you [request at least a large data size](#) for the [ccrz.ccApiOrder.ENTITYNAME](#) key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

To request these related lists even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'E_OrderShipments__r'}
    }
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

**CC Order Shipment**

The order that the shipment applies to.



Usage	Field Name	Data Type
Input on object	ccrz__OrderShipment__c	Master-Detail Relationship ( <a href="#">ccrz__E_OrderShipment__c</a> )
Input or output with API (with any data size)	orderShipment	ID

**CC Order Shipment Item Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

**Order Shipment Item ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__OrderShipmentItemId__c	Text (128)
Input or output with API (with any data size)	orderShipmentItemId	String

**Quantity**

The quantity for the item in the shipment.

Usage	Field Name	Data Type
Input on object	ccrz__Quantity__c	Number (18, 0)
Input or output with API (with any data size)	quantity	Integer

**SKU**

The product SKU for the item in the shipment.

Usage	Field Name	Data Type
Input on object	ccrz__SKU__c	Text (100)
Input or output with API (with any data size)	sku	String

## ccrz\_\_E\_OrderTerm\_\_c

Represents the terms and conditions that a buyer acknowledges and accepts during checkout.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** `ccrz.ccApiOrder`

**Fetch sizing:** The `ccrz.ccApiOrder.fetch` method returns the `orderTermsS` related list when you [request at least a large data size](#) for the `ccrz.ccApiOrder.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'E_OrderShipments__r'}
    }
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

### Fields

#### CC Order Term Auto Number

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number

Usage	Field Name	Data Type
Input or output with API (with any data size)	sfdcName	String

**Description**

Explanatory text for the term, which appears below the title on the Checkout page. This value is copied from the `ccrz__Description__c` field on the corresponding `ccrz__E_Term__c` record.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Description__c</code>	Text Area (Rich) (32768)
Input or output with API (with any data size)	description	String

**Order**

The order that the terms apply to.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Order__c</code>	Master-Detail Relationship ( <code>ccrz__E_Order__c</code> )
Input or output with API (with any data size)	order	ID

**Title**

The label or heading for the term, which displays on the Checkout page with a checkbox that a buyer must select before they can continue. This value is copied from the `ccrz__Title__c` field on the corresponding `ccrz__E_Term__c` record.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Title__c</code>	Text (100)
Input or output with API (with any data size)	title	String

SEE ALSO:

[ccrz\\_\\_E\\_Term\\_\\_c](#)

## ccrz\_\_E\_PageLabel\_\_c

Represents a page label record, which defines buyer-facing text on the storefront, such as headings, buttons, and error messages. Each page label record is specific to one storefront and one locale.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiI18N](#)

**Fetch sizing:** The [ccrz.ccApiI18N.fetchPageLabels](#) method returns the `ccrz.ccApiI18N.PAGE_LABELS` output, where each page label includes only the following fields:

- Page Label Name
- ValueRT

## Fields

### Page Label Name

The name of the page label record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	pageLabelName	String

### Page Name

The name of the storefront page where the page label applies. The value `ALL` specifies that the page label applies to all pages, where applicable.

Usage	Field Name	Data Type
Input on object	ccrz__PageName__c	Text (80)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### PageLabelId

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__PageLabelId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Sequence**

Formula that determines the relative priority of the page label record by trying to match a specific page name, storefront, and locale.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Formula (Number)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Storefront**

The storefront where the page label applies. If the value of this field is `Global`, the page label applies to all storefronts.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (50)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**ValueRT**

The text of the page label for a specific storefront, locale, and page name.

Usage	Field Name	Data Type
Input on object	ccrz__ValueRT__c	Text Area (Rich) (32768)
Input or output with API (with any data size)	value	String

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Value	ccrz__Value__c	Text Area (Long) (32768)

SEE ALSO:

[Salesforce Help Customize Storefront Headings and Text with Page Labels](#)

## ccrz\_\_E\_PageLabelI18N\_\_c

Represents a page label translated for a locale that your storefront supports.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

The API doesn't return fields on the `ccrz__E_PageLabelI18N__c` record directly. When the `ccrz.ccApiI18N.fetchPageLabels` method's input map includes the `ccrz.ccApiI18N.LOCALE` key, the method:

1. References `ccrz__E_PageLabelI18N__c` records for the specified locale.
2. Returns `ccrz__E_PageLabel__c` records, translated with the localized data.

## Fields

### CC Page Label

The page label to localize.

Usage	Field Name	Data Type
Input on object	ccrz__PageLabel__c	Master-Detail Relationship ( <a href="#">ccrz__E_PageLabel__c</a> )
Input or output with API (with any data size)	Not applicable	

**CC PageLabel18N Name**

Name for the localized page label record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	Not applicable	

**LangSequence**

Formula that determines the relative priority of the localized page label.

Usage	Field Name	Data Type
Input on object	ccrz__LangSequence__c	Formula (Number)
Input or output with API (with any data size)	Not applicable	

**Language**

ISO code for the locale to localize for.



**Note:** Each locale uses the following codes for languages and regions, separated by an underscore:

- ISO 639-1 language code, such as *en* or *ja*
- ISO 3166-1 Alpha 2 country or region code, such as *US* or *JP*

Usage	Field Name	Data Type
Input on object	ccrz__Language__c	Text (5)
Input or output with API (with any data size)	Not applicable	

**ValueRT**

The localized page label text.

Usage	Field Name	Data Type
Input on object	ccrz__ValueRT__c	Text Area (Rich) (32768)
Input or output with API (with any data size)	Provides the value for the <code>value</code> field when the API queries localized page label records.	

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Value	ccrz__Value__c	Text Area (Long) (32768)

SEE ALSO:

[Salesforce Help Localize Page Labels for an Internationalized Storefront](#)

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

## ccrz\_\_E\_PageSection\_\_c

Represents a page section record, which specifies extra product content—product specifications, related products, documentation, and so on—to show on the Product Detail page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** The API doesn't directly query records of this object.

### Fields

#### CC Page Section Number

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

#### Display Style

By default, the managed package doesn't use this field.



Usage	Field Name	Data Type
Input on object	ccrz__DisplayStyle__c	Picklist
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Display Type**

Whether the page section displays as a tab or widget on the Product Detail page.

Usage	Field Name	Data Type
Input on object	ccrz__DisplayType__c	Picklist
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Grouping**

The tab group name for all tabs that you want to show together.

 **Note:** The Product Detail page supports only *one* group of page section tabs. In the Capricorn Coffee demo storefront, the default tab group name is `tabSec`.

Usage	Field Name	Data Type
Input on object	ccrz__Grouping__c	Text (25)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Location**

Whether the page section displays in the page's Main Content or Right Column.

Usage	Field Name	Data Type
Input on object	ccrz__Location__c	Picklist
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Page Name**

The page, `ProductDetails`, where the page section displays.

 **Note:** Only the Product Detail page supports page sections.

Usage	Field Name	Data Type
Input on object	ccrz__PageName__c	Picklist

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Page Section ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__PageSectionId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Section Filter**

The content to show in the page section. For usage, see:

- [Show a Product's Long Description](#)
- [Show Related Products](#)
- [Make Documentation Available](#)
- [Show Custom Document Types](#)

Usage	Field Name	Data Type
Input on object	ccrz__SectionFilter__c	Picklist (Multi-select)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Section Filter Other**

The content to show in the page section. For usage, see:

- [Show Custom Content](#)

Usage	Field Name	Data Type
Input on object	ccrz__SectionFilterOther__c	Text (50)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Section Name**

The content to show in the page section. For usage, see:

- [Show a Product's Long Description](#)
- [Show Related Products](#)
- [Show Product Specs on the Product Detail Page](#)

- [Make Documentation Available](#)
- [Show Custom Document Types](#)
- [Show Shipping Information](#)
- [Show Custom Content](#)

Usage	Field Name	Data Type
Input on object	ccrz__SectionName__c	Picklist
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Sequence

The order of the tab or widget among other tab or widget page sections with the same Location and Display Type values. Lower Sequence values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (5, 0)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Storefront

The storefront where the page section applies.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (50)
Input or output with API (with any data size)	The API doesn't return data for this field.	

SEE ALSO:

[Salesforce Help Guidelines for Creating Page Sections](#)

## ccrz\_\_E\_PriceGroup\_\_c

Determines alternative pricing for accounts outside of account groups and price lists. For example, use a price group for contract pricing, special pricing for industry segments, or other context-specific pricing.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** The API doesn't directly query records of this object.

## Fields

### CC Price Group Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Effective End

Date when the price group is no longer available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveEnd__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Effective Start

Date when the price group is available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveStart__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Enabled**

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Group Name**

The price group's name.

Usage	Field Name	Data Type
Input on object	ccrz__GroupName__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Group Type**

The type of price group: Contract, Upgrade, and so on.

Usage	Field Name	Data Type
Input on object	ccrz__GroupType__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Price Group ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__PriceGroupID__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

## SEE ALSO:

[Salesforce Help Honor Custom Prices Negotiated in a Contract](#)

**ccrz\_\_E\_PriceGroupAccount\_\_c**

Represents the assignment of a price group to an account.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** The API doesn't directly query records of this object.

## Fields

### Account

The account that the price group is assigned to.

Usage	Field Name	Data Type
Input on object	ccrz__Account__c	Master-Detail Relationship ( <a href="#">Account</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

### CC Price Group

The price group assigned to the account.

Usage	Field Name	Data Type
Input on object	ccrz__PriceGroup__c	Master-Detail Relationship ( <a href="#">ccrz__E_PriceGroup__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

### CC Price Group Account Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Effective End**

Date when this assignment is no longer available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveEnd__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Effective Start**

Date when this assignment is available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveStart__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Enabled**

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Price Group Account ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__PriceGroupAccountID__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Sequence**

Order of the price group's assignment to the account among all price groups assigned to the account. Lower Sequence values take priority.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Storefront**

The name of the storefront where this assignment applies.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

SEE ALSO:

[Salesforce Help Honor Custom Prices Negotiated in a Contract](#)

**ccrz\_\_E\_PriceGroupItem\_\_c**

Represents a product included in a price group.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**API Usage**

**Class used with:** The API doesn't directly query records of this object.



## Fields

### CC Price Group

The price group that the product is assigned to.

Usage	Field Name	Data Type
Input on object	ccrz__PriceGroup__c	Master-Detail Relationship ( <a href="#">ccrz__E_PriceGroup__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

### CC Price Group Item Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Effective End

Date when this assignment is no longer available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveEnd__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Effective Start

Date when this assignment is available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveStart__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Price Group Item ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__PriceGroupItemID__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Product**

The product to assign.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Quantity**

The allowed quantity of a product for the price group to be valid. Depending on your implementation, this quantity can represent a maximum or minimum requirement.

Usage	Field Name	Data Type
Input on object	ccrz__Quantity__c	Number (14, 4)
Input or output with API (with any data size)	The API doesn't return data for this field.	

## SEE ALSO:

[Salesforce Help Honor Custom Prices Negotiated in a Contract](#)

**ccrz\_\_E\_PriceGroupPricelist\_\_c**

Represents the assignment of a price list to a price group.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** The API doesn't directly query records of this object.

## Fields

### CC Price Group

The price group that the price list is assigned to.

Usage	Field Name	Data Type
Input on object	ccrz__PriceGroup__c	Master-Detail Relationship ( <a href="#">ccrz__E_PriceGroup__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

### CC Price Group Pricelist Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

### CC Price List

The price list assigned to the price group.

Usage	Field Name	Data Type
Input on object	ccrz__Pricelist__c	Lookup Relationship ( <a href="#">ccrz__E_PriceList__c</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Effective End**

Date when this assignment is no longer available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveEnd__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Effective Start**

Date when this assignment is available.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveStart__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Enabled**

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Price Group Pricelist ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__PriceGroupPricelistID__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Sequence**

Order of the price list's assignment to the price group among all price lists assigned to the price group. Lower Sequence values take priority.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Storefront**

The name of the storefront where this assignment applies.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

SEE ALSO:

[Salesforce Help Honor Custom Prices Negotiated in a Contract](#)

## ccrz\_\_E\_Pricelist\_\_c

Represents a price list record, which assigns prices to products for a particular account group. Each price list can apply to only one currency.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiPriceList](#)

**Fetch sizing:** To specify which fields the [ccrz.ccApiPriceList.fetch](#) method returns for each price list record, [request a specific data size](#) for the `ccrz.ccApiPriceList.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
```

```

    }
}

```

The reference for each field indicates which data size the field is returned with.

## Fields

### Currency ISO Code

The ISO 4217 currency code for the price list, such as *USD* or *JPY*.

Usage	Field Name	Data Type
Input on object	ccrz__CurrencyISOCODE__c	Text (10)
Input or output with API (with <b>SZ_S</b> or larger)	currencyISOCODE	String

### Customer Type Code

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__CustomerTypeCode__c	Text (10)
Input or output with API (with <b>SZ_XL</b> )	customerTypeCode	String

### Data ID

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccrzApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	ccrz__DataId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Desc

An internal-facing description for the price list.

Usage	Field Name	Data Type
Input on object	ccrz__CustomerTypeCode__c	Text (255)
Input or output with API (with <b>SZ_XL</b> )	desc	String

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with <b>SZ_M</b> or larger)	enabled	Boolean

**End Date**

The last day when the price list is valid on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with <b>SZ_M</b> or larger)	endDate	Date

**Name**

A unique name for the price list record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Pricelist ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__PricelistId__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	pricelistId	String

**Seller**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__Seller__c	Lookup Relationship ( <a href="#">ccrz__E_Seller__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	seller	ID

**Start Date**

The first day when the price list is valid on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with <b>SZ_M</b> or larger)	startDate	Date

**Storefront**

The storefront where the price list applies.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Picklist (Multi-select)
Input or output with API (with <b>SZ_S</b> or larger)	storefront	String

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Supplier Key	ccrz__SupplierKey__c	Text (255)

SEE ALSO:

[Salesforce Help Manage Prices for Storefront Products with Price Lists](#)

**ccrz\_\_E\_PricelistItem\_\_c**

Represents a specific price for a product in a price list.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiPriceList](#)

**Fetch sizing:** To specify which fields the [ccrz.ccApiPriceList.fetch](#) or [ccrz.ccApiPriceList.fetchPriceListItems](#) method returns for each price list item record, [request a specific data size](#) for the `ccrz.ccApiPriceList.PLI_ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiPriceList.ENTITYNAME_PLI => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Attribute Pricing

JSON data for attribute pricing.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AttributePricing__c</code>	Text Area (Long) (32000)
Input or output with API (with <b>SZ_XL</b> )	<code>attributePricing</code>	String

### End Date

The date when the price list item is no longer valid. For example, if End Date is *12/31/2099*, then December 30, 2099 is the final day that the price for this price list item is valid.

Usage	Field Name	Data Type
Input on object	<code>ccrz__EndDate__c</code>	Date
Input or output with API (with <b>SZ_L</b> or larger)	<code>endDate</code>	Date

### Item Type

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ItemType__c</code>	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	<code>itemType</code>	String

**Max Qty**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__MaxQty__c	Number (18, 0)
Input or output with API (with <b>SZ_L</b> or larger)	maxQty	Integer

**Min Qty**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__MinQty__c	Number (18, 0)
Input or output with API (with <b>SZ_L</b> or larger)	minQty	Integer

**Price**

The price for the product in the price list.

Usage	Field Name	Data Type
Input on object	ccrz__Price__c	Currency (16, 2)
Input or output with API (with <b>SZ_S</b> or larger)	price	Decimal

**Pricelist**

The price for the product in the price list.

Usage	Field Name	Data Type
Input on object	ccrz__Pricelist__c	Master-Detail Relationship ( <a href="#">ccrz__E_PriceList__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	pricelist	ID
Input or output with API (with <b>SZ_M</b> or larger)	pricelistR	Map<String, Object>, which contains the following <a href="#">ccrz__E_PriceList__c</a> fields: <ul style="list-style-type: none"> <li>• currencyISOCode</li> <li>• seller</li> <li>• sfdcName</li> </ul>

**Pricelist Item Id**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__PricelistItemId__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	pricelistItemId	String

**PricelistItemAutoNumber**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Pricing Tiers**

JSON data for tiered pricing. For more information about tiered pricing, see [https://help.salesforce.com/articleView?id=sf.b2b\\_commerce\\_tiered\\_pricing.htm](https://help.salesforce.com/articleView?id=sf.b2b_commerce_tiered_pricing.htm).

Usage	Field Name	Data Type
Input on object	ccrz__PricingTiers__c	Text Area (Long) (32766)
Input or output with API (with <b>SZ_XL</b> )	pricingTiers	String

**Product**

The product assigned a price in the price list.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Lookup Relationship (ccrz__E_Product__c)
Input or output with API (with <b>SZ_S</b> or larger)	product	ID
Input or output with API (with <b>SZ_M</b> or larger)	productR	Map<String, Object>, which contains the following ccrz__E_Product__c fields: <ul style="list-style-type: none"> <li>sku</li> </ul>

**Recurring Price**

Whether the price assigned in this price list item is recurring (selected) or the upfront price (not selected) in a subscription.

Usage	Field Name	Data Type
Input on object	ccrz__RecurringPrice__c	Checkbox
Input or output with API (with <b>SZ_S</b> )	pricingTiers	Boolean

**Start Date**

The date when the price list item is valid.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with <b>SZ_L</b> or larger)	startDate	Date

**SubProdTerm**

The subscription term that the price list item applies to.

Usage	Field Name	Data Type
Input on object	ccrz__SubProdTerm__c	Lookup Relationship ( <a href="#">ccrz__E_SubProdTerm__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	subProdTerm	ID

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Unit Of Measure	ccrz__UnitOfMeasure__c	Text (255)

SEE ALSO:

[Salesforce Help Manage Prices for Storefront Products with Price Lists](#)

**ccrz\_\_E\_Product\_\_c**

Represents a product record for the storefront.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiProduct`

**Field sizing:** To specify which fields the API returns for each product record, [request a specific data size](#) for the `ccrz.ccApiProduct.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Air Flag

Whether the product allows an air-based shipping option. To implement this shipping option:

1. Define a shipping rate for products with air-based shipping. See [Specify Shipping Rates for a Product](#).
2. Ensure that you allow air-based shipping options during checkout. By default, the Checkout page provides an air-based shipping option only when the cart contains a product that has `Air Flag` enabled. To customize how the Checkout page provides shipping options, override the default `ccrz.cc_api_ShippingAndHandling.getShippingOptions` extension point method.

You can also show a product's shipping information in a [page section on the Product Detail page](#).



**Note:** This field is available only in B2B Commerce for Visualforce Winter '21 (version 4.13) or later.

Usage	Field Name	Data Type
Input on object	<code>ccrz__GroundFlag__c</code>	Checkbox
Input or output with API (with <b>SZ_XL</b> )	<code>groundFlag</code>	Boolean

### Alternate Name

An optional longer name that, if specified, takes precedence over the standard `Product Name` field.

Usage	Field Name	Data Type
Input on object	ccrz__AlternateName__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	alternateName	String

**Average Rating**

By default, the managed package doesn't use this field, and doesn't provide a rating system. To use product ratings, integrate an external rating service.

Usage	Field Name	Data Type
Input on object	ccrz__AverageRating__c	Formula (Number)
Input or output with API (with <b>SZ_L</b> or larger)	averageRating	Decimal

**Cooler Pak Flag**

A shipping option for the product, which you can show on the storefront by using a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__CoolerpakFlag__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	coolerpakFlag	Boolean

**Dry Ice Flag**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__DryiceFlag__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	dryiceFlag	Boolean

**Dry Ice Weight**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__DryiceWeight__c	Number (12, 2)
Input or output with API (with <b>SZ_XL</b> )	dryiceWeight	Decimal

**Employment Level**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__Employment_Level__c	Picklist
Input or output with API (with <b>SZ_XL</b> )	employmentLevel	String

**End Date**

The last day that the product is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with <b>SZ_L</b> or larger)	endDate	Date

**Exclude From Product Index**

Don't build a product index record for this product. The product isn't shown in results on the Product List page, and doesn't appear in the category tree. The product still has a Product Detail page, and you can link to the product from a promotion or a Related Products page section.

 **Note:** This field is available only in B2B Commerce for Visualforce Spring '20 (version 4.12) or later.

Usage	Field Name	Data Type
Input on object	ccrz__ExcludeFromProductIndex__c	Checkbox
Input or output with API	The API doesn't return data for this field.	

**Filter Data**

The name of a spec definition and the product's spec value, joined by a vertical bar character. This value is specified automatically when the [product spec index is refreshed](#).

Usage	Field Name	Data Type
Input on object	ccrz__FilterData__c	Text Area (Long) (32768)
Input or output with API (with <b>SZ_XL</b> )	filterData	String

**Ground Flag**

Whether the product allows a ground-based shipping option. To implement this shipping option:

- Define a shipping rate for products with ground-based shipping. See [Specify Shipping Rates for a Product](#)

You can also show a product's shipping information in a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__GroundFlag__c	Checkbox

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_XL</b> )	groundFlag	Boolean

**Hazardous Flag**

A shipping option for the product, which you can show on the storefront by using a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__HazardousFlag__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	hazardousFlag	Boolean

**Hazardous Outside48 Flag**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__HazardousOutside48Flag__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	hazardousOutside48Flag	Decimal

**Inventory Type**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__InventoryType__c	Picklist
Input or output with API (with <b>SZ_L</b> or larger)	inventoryType	String

**Lead Time**

Delivery lead time for the product.

Usage	Field Name	Data Type
Input on object	ccrz__LeadTime__c	Number (18, 0)
Input or output with API (with <b>SZ_XL</b> )	leadTime	Integer

**LongDescRT**

A longer description for the product, which you can show on the storefront by using a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__LongDescRT__c	Text Area (Rich) (5000)



Usage	Field Name	Data Type
Input or output with API (with <b>SZ_L</b> or larger)	longDesc	String

### Number of Reviews

By default, the managed package doesn't use this field, and doesn't provide a rating system. To use product ratings, integrate an external rating service.

Usage	Field Name	Data Type
Input on object	ccrz__NumberOfReviews__c	Roll-Up Summary (COUNT ccrz__E_Review__c)
Input or output with API (with <b>SZ_XL</b> )	numberOfReviews	Integer

### Overnight Flag

A shipping option for the product, which you can show on the storefront by using a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__OvernightFlag__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	overnightFlag	Boolean

### Parent Product

Returned only when you include `ccrz.ccrzApiProduct.PARAM_INCLUDE_PRICING => true` in the input map. Used for implementations of [Set Up Attribute-Based Products](#)

### Primary Attribute

See [Set Up Attribute-Based Products](#)

Usage	Field Name	Data Type
Input on object	ccrz__PrimaryAttr__c	Lookup Relationship (ccrz__E_Attribute__c)
Input or output with API (with <b>SZ_M</b> or larger)	primaryAttr	ID

### Product ID

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__ProductId__c	Text (128)
Input or output with API (with <b>SZ_XL</b> )	productId	String

**Product Index Status**

The status of the product's associated [product index records](#).

Usage	Field Name	Data Type
Input on object	ccrz__ProductIndexStatus__c	Text (255)
Input or output with API	The API doesn't return data for this field.	

**Product Name**

The name that buyers see for the product on your storefront.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Product Status**

The product's status in your storefront catalog, which [determines whether buyers can see the product and add it to a cart](#).

Usage	Field Name	Data Type
Input on object	ccrz__ProductStatus__c	Picklist
Input or output with API (with <b>SZ_S</b> or larger)	productStatus	String

**Product Type**

The type of product, which [determines its relationship with other products and how the product appears on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__ProductType__c	Picklist
Input or output with API (with <b>SZ_S</b> or larger)	productType	String

**Product Weight**

A shipping option for the product, which you can show on the storefront by using a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__ProductWeight__c	Number (12, 2)
Input or output with API (with <b>SZ_XL</b> )	productWeight	Decimal

**Quantity per Unit**

The product's purchasable quantities.

Usage	Field Name	Data Type
Input on object	ccrz__Quantityperunit__c	Number (12,6)
Input or output with API (with <b>SZ_L</b> or larger)	quantityperunit	Decimal

**Required Item Dialog Text**

Text to show in a window when a buyer adds this product to the cart, but the product requires other items to function correctly.

Usage	Field Name	Data Type
Input on object	ccrz__RequiredItemDialogText__c	Text Area (Long) (2000)
Input or output with API (with <b>SZ_XL</b> )	requiredItemDialogText	String

**Review Flag**

By default, the managed package doesn't use this field, and doesn't provide a rating system. To use product ratings, integrate an external rating service.

Usage	Field Name	Data Type
Input on object	ccrz__ReviewFlag__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	reviewFlag	Boolean

**Secondary Attribute**

See [Set Up Attribute-Based Products](#)

**SEO Description**

Text for the content attribute of the `<meta name="description">` and `<meta property="og:description">` tags in the `<head>` of the bot-enabled Product Detail page. This description appears in external search engine results.

Usage	Field Name	Data Type
Input on object	ccrz__SEODescription__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	SEODescription	String

**SEO ID**

A [unique, URL-safe string](#) (up to 255 characters) for the product's friendly URL path.

 **Tip:** The SEO ID field doesn't support the following special characters:

!	*	'	(	)	@	&
---	---	---	---	---	---	---

;	:	+	=	\$	,	.
/	?	%	#	[	]	

Usage	Field Name	Data Type
Input on object	ccrz__SEOId__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	SEOId	String

**SEO Keywords**

A [unique, URL-safe string](#) (up to 255 characters) for the product's friendly URL path.

Usage	Field Name	Data Type
Input on object	ccrz__SEOKeywords__c	Text (150)
Input or output with API (with <b>SZ_L</b> or larger)	SEOKeywords	String

**SEO Title**

A value (up to 255 characters) that appears in the `<title>` in the `<head>` of the bot-enabled Product Detail page.



**Note:** The title appears in external search engine results, the browser's tab or window title, and bookmarks.

Usage	Field Name	Data Type
Input on object	ccrz__SEOTitle__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	SEOTitle	String

**Service Flag**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__ServiceFlag__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	serviceFlag	Boolean

**Ship Separately**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__ShipSeparately__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	shipSeparately	Boolean

**Shipped Individually**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__ShippedIndividually__c	Checkbox
Input or output with API (with <b>SZ_XL</b> )	shippedIndividually	Boolean

**Shipping Weight (lbs)**

A shipping option for the product, which you can show on the storefront by using a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__ShippingWeight__c	Number (12, 2)
Input or output with API (with <b>SZ_XL</b> )	shippingWeight	Decimal

**ShortDescRT**

A short description (up to 2,000 characters) for the product.



**Note:** The short description appears by default throughout your storefront, including:

- On the Product Detail page
- When the product is a result on the Product List page
- When the product is shown in the Featured Products or Spotlight Products widget on the Home page

Usage	Field Name	Data Type
Input on object	ccrz__ShortDescRT__c	Text Area (Rich) (2000)
Input or output with API (with <b>SZ_M</b> or larger)	shortDesc	String

**SKU**

A unique stock unit ID for the product in your inventory system.



**Tip:**

- Avoid 15- or 18-digit alphanumeric values for a SKU. This format matches the standard Salesforce record ID format, which can result in unexpected behavior on the storefront.
- The SKU field doesn't support the following special characters:

!	*	'	(	)	@	&
;	:	+	=	\$	,	.
/	?	%	#	[	]	

Usage	Field Name	Data Type
Input on object	ccrz__SKU__c	Text (100)
Input or output with API (with <b>SZ_S</b> or larger)	SKU	String

**Start Date**

The first day that the product is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with <b>SZ_S</b> or larger)	endDate	Date

**Storefront**

The storefront where the product is available.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Picklist (Multi-select)
Input or output with API (with <b>SZ_S</b> or larger)	storefront	Date

**Supersede Product**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__SupersedeProduct__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_L</b> or larger)	supersedeProduct	ID

**Tax Rate ID**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__TaxRateId__c	Text (128)
Input or output with API (with <b>SZ_L</b> or larger)	taxRateId	String

**Taxable**

Whether the product is eligible for tax calculation during checkout.

Usage	Field Name	Data Type
Input on object	ccrz__Taxable__c	Checkbox
Input or output with API (with <b>SZ_L</b> or larger)	taxable	Boolean

**Tertiary Attribute**

See [Set Up Attribute-Based Products](#)

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Availability Message	ccrz__AvailabilityMessage__c	Text (85)
Available Subscription Frequencies	ccrz__AvailableSubscriptionFrequencies__c	Picklist (Multi-select)
Industry	ccrz__Industry__c	Picklist
Is Subscription Auto Renewal	ccrz__IsSubscriptionAutoRenewal__c	Checkbox
Is Subscription Available	ccrz__IsSubscriptionAvailable__c	Checkbox
Is Subscription Only	ccrz__IsSubscriptionOnly__c	Checkbox
LongDesc	ccrz__LongDesc__c	Text Area (Long) (5000)
Product Index Is Current	ccrz__ProductIndexIsCurrent__c	Checkbox
Sequence	ccrz__Sequence__c	Number (8, 0)
ShortDesc	ccrz__ShortDesc__c	Text Area (Long) (2000)
Subscription Disclaimer	ccrz__SubscriptionDisclaimer__c	Text Area (Long) (32768)
Subscription Duration	ccrz__SubscriptionDuration__c	Number (18, 0)
Subscription Frequency	ccrz__SubscriptionFrequency__c	Number (4, 0)

Field Label	Field Name	Data Type
Unit of Measure	ccrz__UnitOfMeasure__c	Picklist
View in Portal Site	ccrz__PortalSiteURL__c	Formula (Text)

## Fields Returned Only by an API Call

The `ccrz.ccApiProduct.fetch` method returns the following fields but the fields don't have a corresponding input on the `ccrz__E_Product__c` object.

### price

Decimal returned with **SZ\_S** or larger, and returned only when you include

`ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true` in the input map. More pricing data is returned in the `ccrz.ccApiProduct.PRODUCTPRICINGDATA` key.

### showNewSubscriptionSelection

Boolean returned with **SZ\_S** or larger, and returned only when you include

`ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true` in the input map.

## Related Lists

A record of the `ccrz__E_Product__c` object can have multiple related lists that relate the product to other types of records, such as product media or composite products. The `ccrz.ccApiProduct.fetch` method can return data from these related records, depending on the requested data size in the input map.

The reference for each related list indicates which data size the list is returned with by default. You can also use the [ccrz.ccApi.SZ\\_REL flag in your input map](#) to request a particular list, regardless of requested data size:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'CompositeProducts__r'}
    }
}
```

### compositeProductsS

List<Map<String, Object>> of `ccrz__E_CompositeProduct__c`.

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger
<code>ccrz.ccApi.SZ_REL</code> request value	CompositeProducts__r

### EProductGuidesS

List<Map<String, Object>> of `ccrz__E_ProductGuide__c`.

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger



Consideration	Value
ccrz.ccApi.SZ_REL request value	E_ProductGuides__r

**EProductMedias**

List<Map<String, Object>> of [ccrz\\_\\_E\\_ProductMedia\\_\\_c](#).

Consideration	Value
Default data size returned with	<b>SZ_M</b> or larger
ccrz.ccApi.SZ_REL request value	E_ProductMedias__r

**EProductTabs**

List<Map<String, Object>> of [ccrz\\_\\_E\\_ProductTab\\_\\_c](#).

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger
ccrz.ccApi.SZ_REL request value	E_ProductTabs__r

**productInventoryItems**

List<Map<String, Object>> of [ccrz\\_\\_E\\_ProductInventoryItem\\_\\_c](#).

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger
ccrz.ccApi.SZ_REL request value	Product_Inventory_Items__r

**productSpecIndicies**

List<Map<String, Object>> of [ccrz\\_\\_E\\_ProductSpecIndex\\_\\_c](#).

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger
ccrz.ccApi.SZ_REL request value	Product_Spec_Indicies__r

**productSpecs**

List<Map<String, Object>> of [ccrz\\_\\_E\\_ProductSpec\\_\\_c](#).

Consideration	Value
Default data size returned with	<b>SZ_L</b> or larger
ccrz.ccApi.SZ_REL request value	Product_Specs__r

**subProdTerms**

List<Map<String, Object>> of `ccrz__E_SubProdTerm__c`.

Consideration	Value
Default data size returned with	<b>SZ_M</b> or larger
<code>ccrz.ccApi.SZ_REL</code> request value	SubProdTerms__r

SEE ALSO:

[Salesforce Help Create a Product for Your Storefront](#)

[Salesforce Help Specify SEO Metadata for Products and Categories](#)

## ccrz\_\_E\_ProductCategory\_\_c

Represents the assignment of a product to a category.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiCategory`

**Fetch sizing:** The `ccrz.ccApiCategory.fetch` method returns the `productCategoriesS` related list when you request at least a large data size for the `ccrz.ccApiCategory.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

To request this related list even with a smaller data size, add the `ccrz.ccApi.SZ_REL` key to your input map:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'Product_Categories__r'}
    }
}
```

```
}
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

 **Note:** On a storefront, a category can show at most 2,500 products. If a category has more than 2,500 products assigned to it, only the first 2,500 products are shown.

### Category

The category assigned to the product.

Usage	Field Name	Data Type
Input on object	ccrz__Category__c	Master-Detail Relationship ( <a href="#">ccrz__E_Category__c</a> )
Input or output with API (with any data size)	category	ID

### End Date

The last day that the product's category assignment is active on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with any data size)	endDate	Date

### Product

The product with the assigned category.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Master-Detail Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	product	ID

### Product Category Id

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__ProductCategoryId__c	Text (128)

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**ProductCatAutoNumber**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

**Sequence**

Order of the product within the category, relative to other products assigned to the same category. Lower values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with any data size)	sequence	Integer

**Start Date**

The first day that the product's category assignment is active on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with any data size)	startDate	Date

**Canonical Path**

Whether the assigned category is the authoritative category for search engine optimization. For usage, see

[https://help.salesforce.com/articleView?id=b2b\\_commerce\\_seo\\_product\\_category.htm](https://help.salesforce.com/articleView?id=b2b_commerce_seo_product_category.htm).

Usage	Field Name	Data Type
Input on object	ccrz__IsCanonicalPath__c	Checkbox
Input or output with API (with any data size)	isCanonicalPath	Boolean

## ccrz\_\_E\_ProductGuide\_\_c

Represents a product guide used for a dynamic kit.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** `ccrz.ccApiProduct`

**Fetch sizing:** The `ccrz.ccApiProduct.fetch` method returns the `EProductGuidesS` related list when you [request at least a large data size](#) for the `ccrz.ccApiProduct.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'E_ProductGuides__r'}
    }
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

### Fields

#### Display Type

Whether a buyer can select one or many products in the group and whether a buyer can modify quantities:

#### Single-Select

A buyer can select only one product from a group, and can't modify the quantity of the selected product before checkout.

#### Single-Select Quantity

A buyer can select only one product from a group, but can modify the quantity of the selected product before checkout.

**Multi-Select**

A buyer can select multiple products from a group, and can't modify quantities of selected products before checkout.

**Multi-Select Quantity**

A buyer can select multiple products from a group, but can modify quantities of selected products before checkout.

Usage	Field Name	Data Type
Input on object	ccrz__DisplayType__c	Picklist
Input or output with API (with any data size)	displayType	String

**Product**

The dynamic kit that the product guide applies to.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Master-Detail Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	product	ID

**Product Guide Auto Number**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

**Product Guide Auto Number**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__ProductGuideId__c	Text (10)
Input or output with API (with any data size)	productGuideId	String

**Related Product Group**

The related product group that the product guide applies to.

Usage	Field Name	Data Type
Input on object	ccrz__RelatedProductGroup__c	Lookup Relationship ( <a href="#">ccrz__E_RelatedProductGroup__c</a> )
Input or output with API (with any data size)	relatedProductGroup	ID

**Required**

Whether a buyer must select a product from the product group.

Usage	Field Name	Data Type
Input on object	ccrz__Required__c	Checkbox
Input or output with API (with any data size)	required	Boolean

**Sequence**

The order of the product guide among all product guides shown on a dynamic kit's Product Detail page. Lower Sequence values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (18, 0)
Input or output with API (with any data size)	sequence	Integer

SEE ALSO:

[Salesforce Help Create a Dynamic Kit](#)

## ccrz\_\_E\_ProductIndex\_\_c

Represents a flat data structure for one product assigned to one category and one locale. A product index record includes the product's pricing information and searchable data, such as SKU, short and long description, and SEO keywords. B2B Commerce for Visualforce references product index records for quickly populating content on the Product List page version 2.0.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiProductIndex`

**Field sizing:** To specify which fields the API returns for each product index record, [request a specific data size](#) for the `ccrz.ccApiProductIndex.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProductIndex.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Active

Whether this product index record is up-to-date and available for the Product List page to reference. This field supports product index batch jobs. Records that the Product List page currently uses [remain active while the batch job creates updated records](#).

Usage	Field Name	Data Type
Input on object	<code>ccrz__Active__c</code>	Checkbox
Input or output with API (with <b>SZ_L</b> or larger)	<code>active</code>	Boolean

### Category

The category that the product is assigned to.

 **Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Category__c</code>	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	<code>category</code>	ID

### Category End Date

The last day that the category is available on the storefront.

 **Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.



Usage	Field Name	Data Type
Input on object	ccrz__CategoryEndDate__c	Date
Input or output with API (with <b>SZ_L</b> or larger)	categoryEndDate	Date

**Category Start Date**

The first day that the category is available on the storefront.



**Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.

Usage	Field Name	Data Type
Input on object	ccrz__CategoryStartDate__c	Date
Input or output with API (with <b>SZ_M</b> or larger)	categoryStartDate	Date

**CC Product Index Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with <b>SZ_S</b> or larger)	sfid	String

**Filter Data**

A spec and product spec value.

Usage	Field Name	Data Type
Input on object	ccrz__FilterData__c	Text Area (Long) (32768)
Input or output with API	The API doesn't return data for this field.	

**Index Currency**

The currency code for the index price. When you use a [primary index for all currencies](#), this field's value is GLOBAL and doesn't specify a currency.

Usage	Field Name	Data Type
Input on object	ccrz__IndexCurrency__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	indexCurrency	String


**Index Price**

The lowest price for the product, based on all price lists assigned to the product. This field is used only for [sorting on the Product List Page](#), and doesn't apply to any other pricing functionality.

Usage	Field Name	Data Type
Input on object	ccrz__IndexPrice__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	indexPrice	String

**Index Type**

*U0VBUKNI*, the Base64-encoded value for the string *Search*.


 **Note:** B2B Commerce for Visualforce requires this format for identifying products assigned to multiple categories and ensuring that products appear only once in a search. When a product index record specifies *U0VBUKNI* for its Index Type, that product index is the record returned in search results.

 **Warning:** If you extend this field to support other values, make sure that you leave the default value *U0VBUKNI* unchanged.

Usage	Field Name	Data Type
Input on object	ccrz__IndexType__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	indexType	String

**Last Index**

Whether this product index record is the final record created for all the price lists that apply to a product.

 **Note:** The total number of entitled price lists can sometimes exceed the available length of the Price Lists field. In that case, the product index batch job creates another record for the overflow price lists, and sets this field on the overflow record.

Usage	Field Name	Data Type
Input on object	ccrz__LastIndex__c	Checkbox
Input or output with API (with <b>SZ_L</b> or larger)	lastIndex	Boolean

**Locale**

The locale that the product index record applies to.

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (25)
Input or output with API (with <b>SZ_L</b> or larger)	locale	String

**Long Description**

A long description for the product.

Usage	Field Name	Data Type
Input on object	ccrz__LongDescription__c	Text Area (Long) (5000)
Input or output with API	The API doesn't return data for this field.	

**Price Lists**

Price list IDs, separated by a space, that are assigned to the product.

Usage	Field Name	Data Type
Input on object	ccrz__PriceLists__c	Text Area (Long) (131072)
Input or output with API (with <b>SZ_L</b> or larger)	priceLists	String

**Product**

The product that the product index record applies to.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	product	ID

**Product Category**

The category assignment for the product. The Product List page uses this reference to [sort products by the Sequence value of each product's CC Product Category related record](#).

Usage	Field Name	Data Type
Input on object	ccrz__ProductCategory__c	Lookup Relationship ( <a href="#">ccrz__E_ProductCategory__c</a> )
Input or output with API (with <b>SZ_M</b> or larger)	productCategory	ID

**Product Category End Date**

The last day that the product's category assignment is active on the storefront.

 **Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.

Usage	Field Name	Data Type
Input on object	ccrz__ProductCategoryEndDate__c	Date
Input or output with API (with <b>SZ_L</b> or larger)	productCategoryEndDate	Date

#### Product Category Sequence

Order of the product within the category, relative to other products assigned to the same category. Lower values display first.

 **Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.

Usage	Field Name	Data Type
Input on object	ccrz__ProductCategorySequence__c	Number (8, 0)
Input or output with API (with <b>SZ_M</b> or larger)	productCategorySequence	Integer

#### Product Category Start Date

The first day that the product's category assignment is active on the storefront.

 **Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.

Usage	Field Name	Data Type
Input on object	ccrz__ProductCategoryStartDate__c	Date
Input or output with API (with <b>SZ_L</b> or larger)	productCategoryStartDate	Date

#### Product End Date

The last day that the product is available on the storefront.

 **Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.

Usage	Field Name	Data Type
Input on object	ccrz__ProductEndDate__c	Date
Input or output with API (with <b>SZ_L</b> or larger)	productEndDate	Date


#### Product Name

The product's name, localized for the product index record's locale.

Usage	Field Name	Data Type
Input on object	ccrz__ProductName__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	productName	String

### Product Start Date

The first day that the product is available on the storefront.

 **Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.

Usage	Field Name	Data Type
Input on object	ccrz__ProductStartDate__c	Date
Input or output with API (with <b>SZ_M</b> or larger)	productStartDate	Date

### Product Status

The product's visibility on the storefront. For supported values, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_create.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_create.htm).

 **Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.

Usage	Field Name	Data Type
Input on object	ccrz__ProductStatus__c	Text (255)
Input or output with API (with <b>SZ_L</b> or larger)	productStatus	String

### SEO Keywords

Comma-separated list of the product's search keywords. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_seo\\_product\\_category.htm](https://help.salesforce.com/articleView?id=b2b_commerce_seo_product_category.htm).

Usage	Field Name	Data Type
Input on object	ccrz__SEOKeywords__c	Text (255)
Input or output with API	The API doesn't return data for this field.	

### Short Description

The product's short description, localized for the product index record's locale.

Usage	Field Name	Data Type
Input on object	ccrz__ShortDescription__c	Text Area (Rich) (2000)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_L</b> or larger)	shortDescription	String

**SKU**

The product's SKU, localized for the product index record's locale.

Usage	Field Name	Data Type
Input on object	ccrz__SKU__c	Text (100)
Input or output with API (with <b>SZ_L</b> or larger)	SKU	String

**Storefront**

The storefront where the product is available.



**Note:** This field is available only in B2B Commerce for Visualforce Summer '19 (version 4.11) or later.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Picklist (Multi-select)
Input or output with API (with <b>SZ_L</b> or larger)	storefront	String

## SEE ALSO:

[ccrz.ccProductIndexBuilder](#)

[Salesforce Help Maintain Product Index Records](#)

**ccrz\_\_E\_ProductInventoryItem\_\_c**

Represents the available inventory quantity for a product.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** The API doesn't directly query records of this object.

Instead, the `ccrz.ccApiProduct.fetch` return data includes the `productInventoryItems` related list when the `ccrz.ccApiProduct.ENTITYNAME` input map key [requests a large data size](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
  }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
    ccrz.ccApi.SZ_REL => New List<String>{'Product_Inventory_Items__r'}
  }
}
```

**Field sizing:** Each record returned in the related list always returns the same fields, regardless of requested data size.

## Fields

### Availability Message RT

A message that generically describes the product's availability. This message can appear on the Product Detail page instead of the product's specific quantity that's available. For usage, see

[https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_inventory.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_inventory.htm).

Usage	Field Name	Data Type
Input on object	<code>ccrz__AvailabilityMessageRT__c</code>	Text Area (Rich) (256)
Input or output with API (with any data size)	<code>availabilityMessage</code>	String

### Inventory Location Code

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AvailabilityMessageRT__c</code>	Text (50)
Input or output with API (with any data size)	<code>inventoryLocationCode</code>	String

**Product Item**

The product that the inventory information applies to.

Usage	Field Name	Data Type
Input on object	ccrz__ProductItem__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	productItem	ID

**ProductInventoryItem Name**

An internal-facing name for the record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfdcName	String

**Qty Available**

The specific inventory quantity that's available for the product.

Usage	Field Name	Data Type
Input on object	ccrz__QtyAvailable__c	Number (18, 0)
Input or output with API (with any data size)	qtyAvailable	Integer

**Status**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__Status__c	Picklist
Input or output with API (with any data size)	status	String

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Availability Message	ccrz__AvailabilityMessage__c	Text Area (255)



## ccrz\_\_E\_ProductItemI18N\_\_c

Represents an internationalized product record for a specific locale.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

The API doesn't return fields on the `ccrz__E_ProductI18N__c` record directly. When the `ccrz.ccApiProduct.fetch` method's input map includes the `ccrz.ccApiProduct.PRODUCTLOCALE` key, the method:

1. References `ccrz__E_ProductI18N__c` records for the specified locale.
2. Returns `ccrz__E_Product__c` records, translated with the localized data.

### Fields

#### Alternate Name

An optional longer name that, if specified, takes precedence over the standard `Product Item I18N Name` field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AlternateName__c</code>	Text (255)
Input or output with API (with any data size)	Not applicable	

#### Locale

A supported storefront locale.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Locale__c</code>	Text (25)
Input or output with API (with any data size)	Not applicable	

**LongDescRT**

A longer localized description for the product, which you can show on the storefront by using a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__LongDescRT__c	Text Area (Rich) (5000)
Input or output with API (with any data size)	Not applicable	

**Product Item**

The product to localize.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Master-Detail Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	Not applicable	

**Product Item I18N Name**

The localized product name for the locale.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	Not applicable	

**SEO ID**

A [unique, URL-safe string](#) (up to 255 characters) for the product's localized friendly URL path. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_seo\\_product\\_category.htm](https://help.salesforce.com/articleView?id=b2b_commerce_seo_product_category.htm).

 **Tip:** The SEO ID field doesn't support the following special characters:

!	*	'	(	)	@	&
;	:	+	=	\$	,	.
/	?	%	#	[	]	

Usage	Field Name	Data Type
Input on object	ccrz__SEOId__c	Text (255)

Usage	Field Name	Data Type
Input or output with API (with any data size)	Not applicable	

**ShortDescRT**

A short description (up to 2,000 characters) for the product.



**Note:** The short description appears by default throughout your storefront, including:

- On the Product Detail page
- When the product is a result on the Product List page
- When the product is shown in the Featured Products or Spotlight Products widget on the Home page

Usage	Field Name	Data Type
Input on object	ccrz__ShortDescRT__c	Text Area (Rich) (2000)
Input or output with API (with any data size)	Not applicable	

**SKU**

The value of the ccrz\_\_SKU\_\_c field from the source Product Item.

Usage	Field Name	Data Type
Input on object	ccrz__SKU__c	Formula (Text)
Input or output with API (with any data size)	Not applicable	

**Unit of Measure**

Localized unit of measure.

Usage	Field Name	Data Type
Input on object	ccrz__UnitOfMeasure__c	Picklist
Input or output with API (with any data size)	Not applicable	

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
LongDesc	ccrz__LongDesc__c	Text Area (Long) (5000)
ShortDesc	ccrz__ShortDesc__c	Text Area (Long) (2000)

## SEE ALSO:

[Salesforce Help Localize Products for an Internationalized Storefront](#)

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

[Salesforce Help Specify SEO Metadata for Products and Categories](#)

## ccrz\_\_E\_ProductMedia\_\_c

Represents an image or documentation file for a product.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiProduct](#)

**Fetch sizing:** The [ccrz.ccApiProduct.fetch](#) method returns the `EProductMedias` related list when you [request at least a medium data size](#) for the `ccrz.ccApiProduct.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map:](#)

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'E_ProductMedias__r'}
    }
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

### AltMessage

The value of an image's `alt` attribute in the page HTML. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_image.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_image.htm).

Usage	Field Name	Data Type
Input on object	<code>ccrz__AltMessage__c</code>	Text (100)
Input or output with API (with any data size)	<code>altMessage</code>	String

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Enabled__c</code>	Checkbox
Input or output with API (with any data size)	<code>enabled</code>	Boolean

### End Date

The last day when the product media is available on the storefront.

Usage	Field Name	Data Type
Input on object	<code>ccrz__EndDate__c</code>	Date
Input or output with API (with any data size)	<code>endDate</code>	Date

### File Path

The path to the image file from the root of a static resource's ZIP file. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_image\\_static\\_resource.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_image_static_resource.htm).

Usage	Field Name	Data Type
Input on object	<code>ccrz__FilePath__c</code>	Text (128)
Input or output with API (with any data size)	<code>filePath</code>	String

**Locale**

The [storefront locale](#) that the image is localized for. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_image\\_static\\_resource.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_image_static_resource.htm).

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (10)
Input or output with API (with any data size)	locale	String

**Media Type**

Whether the media is a product image, product image thumbnail, or product search image. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_image.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_image.htm).



**Note:** You can also use the Media Type field when showing custom document types on the Product Detail page. For more information, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_page\\_product\\_detail\\_documents\\_custom.htm](https://help.salesforce.com/articleView?id=b2b_commerce_page_product_detail_documents_custom.htm).

Usage	Field Name	Data Type
Input on object	ccrz__MediaType__c	Picklist
Input or output with API (with any data size)	mediaType	String

**Product**

The product that the media record belongs to.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Master-Detail Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	product	ID

**Product Media Auto Number**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

**Product Media ID**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	ccrz__ProductMediaId__c	Text (128)
Input or output with API (with any data size)	sfdcName	String

### Product Media Source

Whether the source for the image is a static resource, content delivery network (CDN), or file attachment. For usage, see:

- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_image.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_image.htm)
- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_media\\_source.htm](https://help.salesforce.com/articleView?id=b2b_commerce_media_source.htm)

Usage	Field Name	Data Type
Input on object	ccrz__ProductMediaSource__c	Picklist
Input or output with API (with any data size)	productMediaSource	String

### Product\_Description

A tooltip string that displays when a buyer hovers over an image.

Usage	Field Name	Data Type
Input on object	ccrz__ProductDescription__c	Text Area (Long) (2000)
Input or output with API (with any data size)	productDescription	String

### Sequence

The order of the product image, when Media Type is Alternate Image Thumbnail, among the product's other alternate thumbnail images. Lower sequence values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with any data size)	sequence	Integer

### Start Date

The first day when the product media is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date

Usage	Field Name	Data Type
Input or output with API (with any data size)	startDate	Date

**Static Resource Name**

The name of the static resource that contains the image. For usage, see

[https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_image\\_static\\_resource.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_image_static_resource.htm).

Usage	Field Name	Data Type
Input on object	ccrz__StaticResourceName__c	Text (255)
Input or output with API (with any data size)	staticResourceName	String

**URI**

The fully qualified URL to the image file from a CDN. For usage, see

[https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_image\\_uri.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_image_uri.htm).

Usage	Field Name	Data Type
Input on object	ccrz__URI__c	Text (255)
Input or output with API (with any data size)	uri	String

**SEE ALSO:**

[Salesforce Help Guidelines for Product Images for B2B Commerce for Visualforce](#)

[Salesforce Help Source Options for Storefront Media](#)

**ccrz\_\_E\_ProductSpec\_\_c**

Represents the value of a product's specification, such as Stainless Steel for a spec named Material.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `productSpecs` field of the `ccrz.ccApiProduct.PRODUCTLIST` key returned from `ccrz.ccApiProduct.fetch`

**Field sizing:** By default, `productSpecs` is returned when the input data for `ccrz.ccApiProduct.fetch` specifies a [large data size](#) for `ccrz.ccApiProduct.ENTITYNAME`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'Product_Specs__r'}
    }
}
```

Each product spec in `productSpecs` always includes the same fields, regardless of requested data size.

## Fields

### Product Item

The product that the spec value applies to.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Product__c</code>	Master-Detail Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API	<code>product</code>	ID

### Product Spec Number

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API	<code>sfdcName</code>	String

**ProductID**

Junction of the product and spec definition that the spec value applies to. Combines the `Product Item` and `Spec` fields, separated by a hyphen character.

Usage	Field Name	Data Type
Input on object	ccrz__ProductID__c	Formula (Text)
Input or output with API	productID	String

**Spec**

The specification that the value applies to, such as `Material`.

Usage	Field Name	Data Type
Input on object	ccrz__Spec__c	Master-Detail Relationship ( <a href="#">ccrz__E_Spec__c</a> )
Input or output with API	spec	ID

**SpecValue**

Value of the product's spec, such as `Stainless Steel`.

**Tip:**

- The Spec Value field is case-insensitive. If the spec's filter type is `Checkbox`, the Feature Filtering widget considers any case variation of the same string to be the same spec value. For example, the values `Commercial`, `commercial`, and `COMMERCIAL` are all counted as the same value.
- If the spec's filter type is `Slider`:
  - Make sure that all spec values you create for that spec are numbers with consistent decimal places. For example, the range of values `1.5`, `2.0`, and `3` isn't valid—instead of `3`, enter `3.0`.
  - Avoid specifying negative numbers for any spec values, which can break the default slider functionality.

Usage	Field Name	Data Type
Input on object	ccrz__SpecValue__c	Text (255)
Input or output with API	specValue	String

## SEE ALSO:

[ccrz\\_\\_E\\_Spec\\_\\_c](#)

[Salesforce Help Define Product Specs and Create Product List Page Filters](#)

[Salesforce Help Show Product Specs on the Product Detail Page](#)

**ccrz\_\_E\_ProductSpecIndex\_\_c**

Represents an instance of a single spec value for a particular product, cached for quickly retrieving spec values on the storefront.

## Compatibility

This reference applies to:


Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiSpec](#)

**Field sizing:** To specify which fields the API returns for each product spec index record, [request a specific data size](#) for the `ccrz.ccApiSpec.PSI_ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiSpec.PSI_ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
    }
}
```

 **Note:** The `ccrz.ccApiProduct.fetch` method can also return product spec index records from a subquery. For more information, see [Other API Usage](#).

The reference for each field indicates which data size the field is returned with.

## Fields

### Product

The product that the indexed spec value applies to.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Product__c</code>	Master-Detail Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	<code>product</code>	ID

### Product Spec Index

Name of the product spec index record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_M</b> or larger)	sfdcName	String

**ProductSpecIndexId**

An external ID that concatenates a product and spec IDs.

Usage	Field Name	Data Type
Input on object	ccrz__ProductSpecIndexId__c	Text (50)
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Spec**

The spec that the indexed spec value applies to, such as Material.

Usage	Field Name	Data Type
Input on object	ccrz__Spec__c	Master-Detail Relationship ( <a href="#">ccrz__E_Spec__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	spec	ID
Other input or output with API (with <b>SZ_M</b> or larger)	specR	Map< <b>String</b> , <b>Object</b> >, which specifies other spec information, including locale and display name.

**SpecValue**

The indexed value of the spec, such as Stainless Steel.

Usage	Field Name	Data Type
Input on object	ccrz__SpecValue__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	specValue	String

**Other API Usage**


**Class used with:** `productSpecIndicesS` field of the `ccrz.ccApiProduct.PRODUCTLIST` key returned from `ccrz.ccApiProduct.fetch`

**Field sizing:** By default, `productSpecIndiciesS` is returned when the input data for `ccrz.ccApiProduct.fetch` specifies a [large data size](#) for `ccrz.ccApiProduct.ENTITYNAME`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
  }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
  ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
    ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
    ccrz.ccApi.SZ_REL => New List<String>{'Product_Spec_Indicies__r'}
  }
}
```

 **Note:** Did you notice that `productSpecIndiciesS` and `Product_Spec_Indicies__r` are misspelled? Let's agree that sometimes the best qualities of the things we love are the imperfections that only we can appreciate.

SEE ALSO:

[ccrz\\_\\_E\\_Spec\\_\\_c](#)

[Salesforce Help Define Product Specs and Create Product List Page Filters](#)

[Salesforce Help Refresh Product Spec Index Records](#)

## ccrz\_\_E\_ProductTab\_\_c

Represents custom rich text content to display as a page section on the Product Detail page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** `ccrz.ccApiProduct`

**Fetch sizing:** The `ccrz.ccApiProduct.fetch` method returns the `EProductTabs` related list when you [request at least a large data size](#) for the `ccrz.ccApiProduct.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

To request this related list even with a smaller data size, [add the `ccrz.ccApi.SZ\_REL` key to your input map](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => New List<String>{'E_ProductTabs__r'}
    }
}
```

Each record returned in this list always includes the same fields, regardless of the requested data size.

## Fields

### ContentRT

Rich text-formatted custom product content to show on the Product Detail page.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ContentRT__c</code>	Text Area (Rich) (32000)
Input or output with API (with any data size)	<code>content</code>	String

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Enabled__c</code>	Checkbox
Input or output with API (with any data size)	<code>enabled</code>	Boolean

### End Date

The last day when the product tab is available on the storefront.

Usage	Field Name	Data Type
Input on object	<code>ccrz__EndDate__c</code>	Date
Input or output with API (with any data size)	<code>endDate</code>	Date

**Locale**

The storefront locale where you want the content to be available, such as *en\_US*.

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (25)
Input or output with API (with any data size)	locale	String

**Product**

The product that the custom content applies to.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Master-Detail Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	product	ID

**Product Tab Number**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

**ProductTab ID**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	ccrz__ProductTabId__c	Auto Number
Input or output with API (with any data size)	productTabId	String

**Sequence**

The order of the tab among other custom product tabs assigned to the product. Lower Sequence values appear first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)

Usage	Field Name	Data Type
Input or output with API (with any data size)	sequence	Integer

**Start Date**

The first day when the product tab is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with any data size)	startDate	Date

**Tab**

A key for referencing the custom content in a page section on the Product Detail page.

Usage	Field Name	Data Type
Input on object	ccrz__Tab__c	Picklist
Input or output with API (with any data size)	tab	String

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Content	ccrz__Content__c	Text Area (Long) (32000)

SEE ALSO:

[Salesforce Help Show Custom Content on the Product Detail Page](#)

**ccrz\_\_E\_Promo\_c**

Represents a promotion record for a storefront.

**Compatibility**

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiPromotion`

**Fetch sizing:** An API `Fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

## Fields

### Category

A category that the promotion links to when Link Type is Category. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_category.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_category.htm).

Usage	Field Name	Data Type
Input on object	<code>ccrz__Category__c</code>	Lookup Relationship ( <code>ccrz__E_Category__c</code> )
Input or output with API (with any data size)	<code>category</code>	ID
	<code>categoryR</code>	<code>Map&lt;String, Object&gt;</code> that describes the category.

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Enabled__c</code>	Checkbox
Input or output with API (with any data size)	<code>enabled</code>	Boolean

### End Date

The last day that the promotion is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with any data size)	endDate	Date

**External Link**

The URL that the promotion links to when Link Type is URL. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_url.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_url.htm).

Usage	Field Name	Data Type
Input on object	ccrz__ExternalLink__c	URL
Input or output with API (with any data size)	externalLink	String

**Image Source**

Whether the source for the image is a static resource, content delivery network (CDN), or file attachment. For usage, see:

- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_detail.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_detail.htm)
- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_media\\_source.htm](https://help.salesforce.com/articleView?id=b2b_commerce_media_source.htm)

Usage	Field Name	Data Type
Input on object	ccrz__ImageSource__c	Picklist
Input or output with API (with any data size)	imageSource	String

**Image URI**

The path to the image file from the root of the static resource's ZIP file, or the fully qualified URL to the image file from a CDN. For usage, see:

- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_image\\_static\\_resource.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_image_static_resource.htm)
- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_image\\_uri.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_image_uri.htm)

Usage	Field Name	Data Type
Input on object	ccrz__ImageURI__c	Text (128)
Input or output with API (with any data size)	imageURI	String

**Link Type**

Whether the promotion links to a category, product, or URL. For valid values and use cases, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_detail.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_detail.htm).

Usage	Field Name	Data Type
Input on object	ccrz__LinkType__c	Picklist
Input or output with API (with any data size)	linkType	String

### Location Display Sequence

Display order of the promotion within a group of promotions with the same Location Type.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with any data size)	sequence	Integer

### Location Type

The area of a page where the promotion displays. For valid values and use cases, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_detail.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_detail.htm).

Usage	Field Name	Data Type
Input on object	ccrz__LocationType__c	Picklist
Input or output with API (with any data size)	locationType	String

### Name

An internal-facing name for the record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfdcName	String

### New Window

Whether to open a new browser window for navigating to an external link. This field is valid only when Link Type is URL.

Usage	Field Name	Data Type
Input on object	ccrz__NewWindow__c	Checkbox
Input or output with API (with any data size)	newWindow	Boolean

**Page Location**

The storefront page where the promotion displays. For valid values and use cases, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_detail.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_detail.htm).

Usage	Field Name	Data Type
Input on object	ccrz__PageLocation__c	Picklist
Input or output with API (with any data size)	pageLocation	String

**Product**

A product that the promotion links to when Link Type is Product.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with any data size)	product	ID
	productR	Map<String, Object>, which specifies other product information, including SKU and ID.

**Product Related**

Whether the promotion applies to a product.

Usage	Field Name	Data Type
Input on object	ccrz__ProductRelated__c	Formula (Text)
Input or output with API (with any data size)	productRelated	String

**Start Date**

The first day that the promotion is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with any data size)	startDate	Date

**Static Resource**

The name of the static resource that contains the promotion image. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_image\\_static\\_resource.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_image_static_resource.htm).

Usage	Field Name	Data Type
Input on object	ccrz__StaticResource__c	Text (80)
Input or output with API (with any data size)	staticResource	String

**Storefront**

The first day that the promotion is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StorefrontMS__c	Picklist (Multi-Select)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**True**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__True__c	Formula (Checkbox)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Coupon	ccrz__Coupon__c	Lookup Relationship ( <a href="#">ccrz__E_Coupon__c</a> )
Graphic Type	ccrz__Graphic_Type__c	Picklist
Long Description	ccrz__LongDesc__c	Text Area (Long) (1000)
New Window Property	ccrz__NewWindowProperty__c	Text Area (255)
Script	ccrz__Script__c	Text Area (Long) (32768)
Short Desc	ccrz__ShortDesc__c	Text Area (255)
Splash Rotate	ccrz__SplashRotate__c	Checkbox

Field Label	Field Name	Data Type
Storefront	ccrz__Storefront__c	Text (255)

SEE ALSO:

[Salesforce Help Best Practices When Creating Promotions for Your Storefront](#)

## ccrz\_\_E\_Promol18n\_c

Represents a localized promotion record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

The API doesn't return fields on the `ccrz__E_PromoI18N__c` record directly. When the `ccrz.ccApiPromotion.fetch` method's input map includes the `ccrz.ccApiPromotion.PROMOTIONLOCALE` key, the method:

1. References `ccrz__E_PromoI18N__c` records for the specified locale.
2. Returns `ccrz__E_Promo__c` records, translated with the localized data.

## Fields

### AttachmentId

Salesforce ID of an attached image file when `Image Source` is Attachment. The actual image is attached to the parent `ccrz__E_Promo__c` record.

Usage	Field Name	Data Type
Input on object	ccrz__AttachmentId__c	Text (50)
Input or output with API (with any data size)	Not applicable	

**CC Promotion**

The promotion record to localize.

Usage	Field Name	Data Type
Input on object	ccrz__Promotion__c	Master-Detail Relationship (ccrz__E_Promo__c)
Input or output with API (with any data size)	Not applicable	

**External Link**

The locale-specific URL that the internationalized promotion links to when the parent ccrz\_\_E\_Promo\_\_c record's Link Type is URL.

Usage	Field Name	Data Type
Input on object	ccrz__ExternalLink__c	Text (255)
Input or output with API (with any data size)	Not applicable	

**Image Source**

Whether the source for the internationalized promotion image is a static resource, content delivery network (CDN), or file attachment. For usage, see:

- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_detail.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_detail.htm)
- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_media\\_source.htm](https://help.salesforce.com/articleView?id=b2b_commerce_media_source.htm)

Usage	Field Name	Data Type
Input on object	ccrz__ImageSource__c	Picklist
Input or output with API (with any data size)	Not applicable	

**Image URI**

The path to the image file from the root of the static resource's ZIP file, or the fully qualified URL to the image file from a CDN. For usage, see:

- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_image\\_static\\_resource.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_image_static_resource.htm)
- [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_image\\_uri.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_image_uri.htm)

Usage	Field Name	Data Type
Input on object	ccrz__ImageURI__c	Text (128)
Input or output with API (with any data size)	Not applicable	

**Locale**

A supported storefront locale.

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (128)
Input or output with API (with any data size)	Not applicable	

**Promotion i18n Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	Not applicable	

**Static Resource**

The name of the static resource that contains the internationalized promotion image. For usage, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_promotion\\_image\\_static\\_resource.htm](https://help.salesforce.com/articleView?id=b2b_commerce_promotion_image_static_resource.htm).

Usage	Field Name	Data Type
Input on object	ccrz__StaticResource__c	Date
Input or output with API (with any data size)	Not applicable	

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Description	ccrz__Description__c	Text Area (Rich) (32768)

SEE ALSO:

[Salesforce Help Localize Promotions for an Internationalized Storefront](#)

**ccrz\_\_E\_PromotionAccountGroupFilter\_\_c**

Represents an account group that a promotion record includes or excludes, which targets the promotion at a specific set of users.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** The API doesn't directly query records of this object.

Instead, the `ccrz.ccApiPromotion.fetch` method filters returned promotion records by the account group that this object specifies when the input map specifies `ccrz.ccApiPromotion.FILTERED => true`.

## Fields

### CC Account Group

The account group that the filter applies to.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AccountGroup__c</code>	Lookup Relationship ( <code>ccrz__E_AccountGroup__c</code> )
Input or output with API (with any data size)	Not applicable	

### CC Promotion

The promotion that the filter applies to.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Promotion__c</code>	Master-Detail Relationship ( <code>ccrz__E_Promo__c</code> )
Input or output with API (with any data size)	Not applicable	

### CC Promotion Account Group Filter Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	Not applicable	

**End Date**

The date when this filter is no longer active.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with any data size)	Not applicable	

**Filter Type**

Whether the filter includes or excludes the buyers in the specified account group.

Usage	Field Name	Data Type
Input on object	ccrz__FilterType__c	Picklist
Input or output with API (with any data size)	Not applicable	

**Start Date**

The date when this filter is active.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with any data size)	Not applicable	

SEE ALSO:

[Salesforce Help Target a Promotion at Specific Storefront Users](#)

**ccrz\_\_E\_PublicCache\_\_c**

Represents cached data for content that's publicly available and specific to a storefront, page, and locale.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** The API doesn't directly query records of this object.

The `ccrz.ccApiPublicCache.fetch` method returns data from `ccrz__E_PublicCacheContent__c` records.

## Fields

### Build Group

A value that's shared by multiple public cache records to group them together as part of the same build operation. For example, sometimes building the page label cache from a batch job requires multiple batches to fully rebuild the page label cache. Such a batch job creates multiple public cache records with the same Cache Name and Build Group. By associating multiple public cache records to a single build operation, you can more easily manage all the records related to that build.

Usage	Field Name	Data Type
Input on object	<code>ccrz__BuildGroup__c</code>	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Build Timestamp

Date and time for the build group, which identifies when the cache record was built. For all cache records that belong to the same build group, use the same timestamp.

Usage	Field Name	Data Type
Input on object	<code>ccrz__BuildTimestamp__c</code>	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Cache Label

Usage	Field Name	Data Type
Input on object	<code>ccrz__CacheLabel__c</code>	Text (255)

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Cache Name**

The type of data that the record caches, such as:

- *ccCategoryTree* for cached page label data that `ccrz.ccCategoryCacheBuildJob` builds.
- *ccPageLabel* for cached page label data that `ccrz.ccPageLabelCacheBuildJob` builds.

Usage	Field Name	Data Type
Input on object	ccrz__CacheName__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**CacheId**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__CacheId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**CC Public Cache Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Data ID**

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	ccrz__DataId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Effective End**

Date and time when the cache becomes valid.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveEnd__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Effective Start**

Date and time when the cache is no longer valid.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveStart__c	Date/Time
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Enabled**

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Locale**

Locale for the cached data. When this field is empty, the cached data isn't specific to a locale.

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (10)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Pagename**

Page that the cached data applies to. When this field is empty or `a11`, the cached data applies to all pages.

Usage	Field Name	Data Type
Input on object	ccrz__Pagename__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**SequencePriority**

The priority of [which cache to return](#).

Usage	Field Name	Data Type
Input on object	ccrz__SequencePriority__c	Number (4, 0)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Storefront**

Storefront that the cached data applies to. When this field is empty or `global`, the cached data applies to all storefronts.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**ccrz\_\_E\_PublicCacheContent\_\_c**

Represents the content of a parent public cache record.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**API Usage**

The `ccrz.ccApiPublicCache.CONTENT` return key of the `ccrz.ccApiPublicCache.fetch` method contains the value of this object's Content field.

The API doesn't directly query this object or return other fields on this object.

## Fields

### Cache Content ID

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	ccrz__CacheContentId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### CC Public Cache Content Name

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Content

Content for the public cache record.

Usage	Field Name	Data Type
Input on object	ccrz__Content__c	Text Area (Long) (130000)
Input or output with API (with any data size)	content	String

### Data ID

The managed package only uses this field internally. Your subscriber code's unit tests developed with the `ccrz.ccrzApiTestData` class can also reference this field.

Usage	Field Name	Data Type
Input on object	ccrz__DataId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Parent Cache

The public cache record that the content applies to.

Usage	Field Name	Data Type
Input on object	ccrz__ParentCache__c	Master-Detail Relationship ( <a href="#">ccrz__E_PublicCache__c</a> )

Usage	Field Name	Data Type
Input or output with API (with any data size)	content	String

### Sequence

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (18, 0)
Input or output with API (with any data size)	The API doesn't return data for this field.	

## ccrz\_\_E\_RelatedProduct\_\_c

Represents a relationship between two products, such as specifying that one product in your catalog complements or replaces another product.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiRelatedProduct`

**Fetch sizing:** To specify which fields the `ccrz.ccApiRelatedProduct.fetch` method returns for each account record, request a specific data size for the `ccrz.ccApiRelatedProduct.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiRelatedProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

The reference for each field indicates which data size the field is returned with.



## Fields

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with <b>SZ_S</b> or larger)	enabled	Boolean

### End Date

The day when the product relationship is no longer available.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with <b>SZ_S</b> or larger)	enabled	Boolean

### Product

The primary product that the relationship is for.

Usage	Field Name	Data Type
Input on object	ccrz__Product__c	Master-Detail Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	product	ID

### Related Product

The product to relate to the primary selected product.

Usage	Field Name	Data Type
Input on object	ccrz__RelatedProduct__c	Lookup Relationship ( <a href="#">ccrz__E_Product__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	relatedProduct	ID
	relatedProductR	Map<String, Object> that includes the related product's SKU.
Input or output with API (with <b>SZ_M</b> or larger)	relatedProductR	Map<String, Object> that includes the related product's SKU, product type, unit of measure, and name.

**Related Product Group**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__RelatedProductGroup__c	Lookup Relationship ( <a href="#">ccrz__E_RelatedProductGroup__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	relatedProductGroup	ID

**Related Product ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__RelatedProductId__c	Text (128)
Input or output with API (with <b>SZ_S</b> or larger)	relatedProductId	String

**Related Product Type**

The type of relationship between products. For supported values and use cases, see [Create Relationships Between Products](#).

Usage	Field Name	Data Type
Input on object	ccrz__RelatedProductType__c	Picklist
Input or output with API (with <b>SZ_S</b> or larger)	relatedProductType	String

**RelatedAutoNumber**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Relationship Description**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__RelationshipDescription__c	Text Area (Rich) (1000)

Usage	Field Name	Data Type
Input or output with API (with <b>SZ_S</b> or larger)	relationshipDescription	String

### Sequence

The order of the related product among all the primary product's related products with the same Related Product Type.

 **Note:** In general, lower Sequence values display first. When Related Product Type is set to Accessories or CrossSell, higher Sequence values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8,0)
Input or output with API (with <b>SZ_S</b> or larger)	sequence	Integer

### Start Date

The day when the product relationship is available.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with <b>SZ_S</b> or larger)	startDate	Boolean

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
ShortDesc	ccrz__ShortDesc__c	Text (255)

SEE ALSO:

[Salesforce Help Create Relationships Between Products](#)

## ccrz\_\_E\_RelatedProductGroup\_\_c

This object is available in the managed package, but isn't used for any default functionality.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** The API doesn't directly query records of this object.

## Fields

### GroupId

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__GroupId__c	Text (4)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Related Product Group Name

An internal-facing name for the record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	The API doesn't return data for this field.	

## ccrz\_\_E\_Seller\_\_c

Represents a seller hosted on your storefront whose products are available for search and purchase.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiSeller`

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

## Fields

### Account

The account associated with the seller.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Account__c</code>	Lookup Relationship ( <a href="#">Account</a> )
Input or output with API (with any data size)	<code>account</code>	ID

### Default Representative

A standard user assigned as the seller's representative. This user owns the portion of the buyer's order that the seller is responsible for fulfilling.

Usage	Field Name	Data Type
Input on object	<code>ccrz__DefaultRep__c</code>	Lookup Relationship ( <a href="#">User</a> )
Input or output with API (with any data size)	<code>defaultRep</code>	ID

### Effective End

The final date when the seller is active on the storefront.

Usage	Field Name	Data Type
Input on object	<code>ccrz__EffectiveEnd__c</code>	Date/Time
Input or output with API (with any data size)	<code>effectiveEnd</code>	Datetime

**Effective Start**

The first date when the seller is active on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveStart__c	Date/Time
Input or output with API (with any data size)	effectiveStart	Datetime

**Enabled**

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Location**

Latitude and longitude values, which B2B Commerce for Visualforce uses to determine the seller's distance from a buyer in the Seller Locator widget on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__Location__c	Geolocation
Input or output with API (with any data size)	location	Location

**Logo**

URL for the seller's logo image to show on the storefront. Use https protocol.

Usage	Field Name	Data Type
Input on object	ccrz__Logo__c	URL (255)
Input or output with API (with any data size)	logo	String

**Long Description**

Content for the Seller Detail page.

Usage	Field Name	Data Type
Input on object	ccrz__LongDesc__c	Text Area (Rich) (32768)

Usage	Field Name	Data Type
Input or output with API (with any data size)	longDesc	String

### Search Tokens

Keywords that complement the Seller Name and Location fields as searchable metadata to help buyers find a seller from the Seller Locator widget on the storefront.

 **Tip:** To support buyers who search only by distance in the Seller Locator widget and don't enter any other search criteria, always include the keyword `all`.

Usage	Field Name	Data Type
Input on object	ccrz__SearchTokens__c	Text Area (Long) (32768)
Input or output with API (with any data size)	searchTokens	String

### Seller ID

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__SellerId__c	Text (255)
Input or output with API (with any data size)	sellerId	String

### Seller Name

Name that appears on the Seller List and Seller Detail pages.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfdcName	String

### Short Description

Content for the Seller List page.

Usage	Field Name	Data Type
Input on object	ccrz__ShortDesc__c	Text Area (Rich) (32768)
Input or output with API (with any data size)	shortDesc	String

**Storefront**

Storefront where the seller has products available.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with any data size)	storefront	String

**Website URL**

URL for the seller's external website. Include the `http` or `https` protocol.

Usage	Field Name	Data Type
Input on object	ccrz__WebsiteURL__c	URL (255)
Input or output with API (with any data size)	websiteURL	String

SEE ALSO:

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

**ccrz\_\_E\_SellerI18N\_\_c**

Represents a localized seller record.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**API Usage**

The API doesn't return fields on the `ccrz__E_SellerI18N__c` record directly. When the `ccrz.ccApiSeller.fetch` method's input map includes the `ccrz.ccApiSeller.SELLERLOCALE` key, the method:

1. References `ccrz__E_SellerI18N__c` records for the specified locale.



2. Returns `ccrz__E_Seller__c` records, translated with the localized data.

## Fields

### CC Seller

The seller record to localize.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Seller__c</code>	Master-Detail Relationship ( <code>ccrz__E_Seller__c</code> )
Input or output with API (with any data size)	Not applicable	

### CC Seller I18N Name

An internal-facing name for the record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	Not applicable	

### Effective End

The final date when the internalized seller record is active on the storefront.

Usage	Field Name	Data Type
Input on object	<code>ccrz__EffectiveEnd__c</code>	Date/Time
Input or output with API (with any data size)	Not applicable	

### Effective Start

The first date when the internalized seller record is active on the storefront.

Usage	Field Name	Data Type
Input on object	<code>ccrz__EffectiveStart__c</code>	Date/Time
Input or output with API (with any data size)	Not applicable	

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Locale

ISO code for the locale to localize for.



**Note:** Each locale uses the following codes for languages and regions, separated by an underscore:

- ISO 639-1 language code, such as *en* or *ja*
- ISO 3166-1 Alpha 2 country or region code, such as *US* or *JP*

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (25)
Input or output with API (with any data size)	Not applicable	

### Logo

URL for the internationalized seller's logo image to show on the storefront. Use `https` protocol.

Usage	Field Name	Data Type
Input on object	ccrz__Logo__c	URL (255)
Input or output with API (with any data size)	Not applicable	

### Long Description

Content for the internationalized Seller Detail page.

Usage	Field Name	Data Type
Input on object	ccrz__LongDesc__c	Text Area (Rich) (32768)
Input or output with API (with any data size)	Not applicable	

### Search Tokens

Internationalized keywords that complement the Seller Name and Location fields as searchable metadata to help buyers find a seller from the Seller Locator widget on the storefront.



**Tip:** To support buyers who search only by distance in the Seller Locator widget and don't enter any other search criteria, always include the keyword *a11*.

Usage	Field Name	Data Type
Input on object	ccrz__SearchTokens__c	Text Area (Long) (32768)
Input or output with API (with any data size)	Not applicable	

**SellerI18N ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__SellerI18NId__c	Text (255)
Input or output with API (with any data size)	Not applicable	

**Short Description**

Content for the internationalized Seller List page.

Usage	Field Name	Data Type
Input on object	ccrz__ShortDesc__c	Text Area (Rich) (32768)
Input or output with API (with any data size)	Not applicable	

SEE ALSO:

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)


## ccrz\_\_E\_ShippingRate\_\_c

Represents a shipping cost calculation for a product, based on the product's specified shipping weight, a shipping carrier, and a shipping method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** The CC Shipping Rate object is designed for compatibility with standard checkout, but not line-level independence (LLI) checkout. B2B Commerce for Visualforce doesn't support solutions that use LLI checkout with the CC Shipping Rate object. To set up shipping methods for LLI checkout, extend the `ccrz.cc_api_ShippingAndHandling` extension point class.

## API Usage

**Class used with:** The API doesn't directly query records of this object.

## Fields

### Air

The shipping rate applies to an Air shipping service.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Air__c</code>	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

### CC Shipping Rate Name

The name that appears in the Shipping Method picklist on the storefront's Checkout page.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Currency Code

ISO 4217 currency code, such as *USD* or *JPY*, for the shipping rate.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CurrencyISOCODE__c</code>	Text (10)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Default

Whether the shipping rate is the default selection in the Shipping Method picklist on the storefront's Checkout page.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Default__c</code>	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Ground**

The shipping rate applies to a Ground shipping service.

Usage	Field Name	Data Type
Input on object	ccrz__Ground__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Max Weight**

The maximum product shipping weight that this rate applies to.

Usage	Field Name	Data Type
Input on object	ccrz__MaxWeight__c	Number (18, 2)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Min Weight**

The minimum product shipping weight that this rate applies to.

Usage	Field Name	Data Type
Input on object	ccrz__MinWeight__c	Number (18, 2)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Next Day Air**

The shipping rate applies to a Next Day Air shipping service.

Usage	Field Name	Data Type
Input on object	ccrz__NextDayAir__c	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Sequence**

By default, the managed package doesn't use this field.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Shipper**

The name of the service that fulfills the shipment.

Usage	Field Name	Data Type
Input on object	ccrz__Shipper__c	Text (128)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Shipping Rate**

Cost for the shipping rate in the specified currency.

Usage	Field Name	Data Type
Input on object	ccrz__ShippingRate__c	Currency (18, 2)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**ShippingRateId**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__ShippingRateId__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Storefront**

Storefront where the shipping rate applies.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Picklist (Multi-select)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**SEE ALSO:**

[Salesforce Help Specify Shipping Rates for a Product](#)

[Salesforce Help Specify Shipping Options for Standard Checkout](#)

## ccrz\_\_E\_Spec\_\_c

Represents the definition for a specification, which identifies a type of physical, operational, or performance measurement for a product. The `ccrz__E_ProductSpec__c` object defines a specification value for a particular product.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiSpec`

**Field sizing:** To specify which fields the API returns for each product spec index record, [request a specific data size](#) for the `ccrz.ccApiSpec.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiSpec.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Can be Used for Filtering

Whether a buyer can [filter results on the Product List page by the spec's values](#).

Usage	Field Name	Data Type
Input on object	<code>ccrz__UseForFilter__c</code>	Checkbox
Input or output with API (with <b>SZ_S</b> or larger)	<code>useForFilter</code>	Boolean

### Desc

The managed package only uses this field internally.

Usage	Field Name	Data Type
Input on object	ccrz__Desc__c	Text (255)
Input or output with API (with <b>SZ_M</b> or larger)	desc	String

**Display Name**

The spec name shown in a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__DisplayName__c	Text (80)
Input or output with API (with <b>SZ_M</b> or larger)	displayName	String

**Filter Display Name**

The spec name shown in a [filter on the Product List page](#).

Usage	Field Name	Data Type
Input on object	ccrz__FilterDisplayName__c	Text (80)
Input or output with API (with <b>SZ_M</b> or larger)	filterDisplayName	String

**Filter Type**

Whether the spec's filter renders as a [checkbox or slider on the Product List page](#).

Usage	Field Name	Data Type
Input on object	ccrz__FilterType__c	Picklist
Input or output with API (with <b>SZ_M</b> or larger)	descFilterType	String

**Is Comparable**

Whether the spec is [available for the Product Compare page](#).

Usage	Field Name	Data Type
Input on object	ccrz__IsComparable__c	Checkbox
Input or output with API (with <b>SZ_S</b> or larger)	isComparable	Boolean




**Is Visible in Catalog**

Whether the spec can appear in a [page section on the Product Detail page](#).

Usage	Field Name	Data Type
Input on object	ccrz__IsVisibleInCatalog__c	Checkbox
Input or output with API (with <b>SZ_S</b> or larger)	isComparable	Boolean

**Locale**


The storefront locale where the spec applies.

 **Note:** You can save a spec without specifying a locale, but the spec only functions and displays correctly on the storefront when you specify a locale.

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (25)
Input or output with API (with <b>SZ_M</b> or larger)	locale	String

**Name**

Internal display name for the spec record.

 **Note:** When you [localize a spec record for an internationalized storefront](#), specify the same Name value for each locale-specific spec record. The `ccrz.ccApiSpec.fetch` method queries localized spec records by matching the value of the Name field. Avoid reusing the same spec Name value for more than one spec per locale.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with <b>SZ_S</b> or larger)	sfdcName	String

**Sequence**

An integer that specifies [how prominently the spec displays on the storefront](#). Lower Sequence values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (8, 0)
Input or output with API (with <b>SZ_S</b> or larger)	sequence	Integer

**SpecGroup**

The heading in the [page section on the Product Detail page](#) where the spec displays.

Usage	Field Name	Data Type
Input on object	ccrz__SpecGroup__c	Picklist
Input or output with API (with <b>SZ_M</b> or larger)	specGroup	String

**SpecId**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__SpecID__c	Text (128)
Input or output with API (with <b>SZ_S</b> or larger)	specID	ID

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Select Type	ccrz__SelectType__c	Picklist

SEE ALSO:

[ccrz\\_\\_E\\_ProductSpec\\_\\_c](#)

[Salesforce Help Define Product Specs and Create Product List Page Filters](#)

[Salesforce Help Show Product Specs on the Product Detail Page](#)

**ccrz\_\_E\_SiteIndex\_\_c**

Represents the hashed definition of a single category's friendly URL and its associated breadcrumb path for a specific storefront and locale.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiSiteIndex](#)

**Field sizing:** To specify which fields the API returns for each site index record, [request a specific data size](#) for the `ccrz.ccApiSiteIndex.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiSiteIndex.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

The reference for each field indicates which data size the field is returned with.

## Fields

### Active

Whether this site index record is up-to-date and available for the storefront to reference. This field supports site index batch jobs. Records that the storefront currently uses [remain active while the batch job creates updated records](#).

Usage	Field Name	Data Type
Input on object	<code>ccrz__Active__c</code>	Checkbox
Input or output with API (with <b>SZ_S</b> or larger)	<code>active</code>	Boolean

### Breadcrumb

JSON data that represents the breadcrumb path for the category and any of its parent categories. For example, the following breadcrumb JSON data describes the Medium category in the Capricorn Coffee demo storefront data. The breadcrumb data includes details for the Medium category and its parent category, Coffee.

```
[
  {
    "friendlyUrl":"/Coffee/Medium",
    "category":{
      "sfid":"a0M3h0000007LjGEAU",
      "name":"Medium"
    }
  },
  {
    "category":{
      "name":"Coffee",
      "sfid":"a0M3h0000007LjAEAU"
    }
  },
]
```

```

        "friendlyUrl":"/Coffee"
    }
]

```

For information about how the breadcrumbs appear on the storefront, see [Show the Breadcrumb Widget](#).

Usage	Field Name	Data Type
Input on object	ccrz__Breadcrumb__c	Text Area (Long) (30000)
Input or output with API (with <b>SZ_M</b> or larger)	breadcrumb	String

### Category

The category that the site index record describes.

Usage	Field Name	Data Type
Input on object	ccrz__Category__c	Lookup Relationship ( <a href="#">ccrz__E_Category__c</a> )
Input or output with API (with <b>SZ_S</b> or larger)	category	ID

### CC Site Index Name

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Friendly URL

The category's [friendly URL path](#).

Usage	Field Name	Data Type
Input on object	ccrz__FriendlyURL__c	Text Area (Long) (5000)
Input or output with API (with <b>SZ_S</b> or larger)	friendlyURL	String

### Friendly URL Hash

The result of passing the `Friendly URL` value through the Apex `String.hashCode` method. The hashed value allows for efficiently querying records by friendly URL values.

Usage	Field Name	Data Type
Input on object	ccrz__FriendlyURLHash__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	friendlyURLHash	String

**Locale**

The locale that the site index record applies to.

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (25)
Input or output with API (with <b>SZ_S</b> or larger)	locale	String

**Site Index Id**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__SiteIndexId__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	The API doesn't return data for this field.	

**Storefront**

The storefront that the site index record applies to.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with <b>SZ_S</b> or larger)	storefront	String

## SEE ALSO:

[ccrz.ccSiteIndexBuilder](#)

[Salesforce Help Refresh Storefront Site Index Records](#)

[Salesforce Help Specify SEO Metadata for Products and Categories](#)

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

[Salesforce Help Show the Breadcrumb Widget](#)

**ccrz\_\_E\_StoredPayment\_\_c**

Represents payment details saved in the My Wallet section of the My Account page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiStoredPayment](#)

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

## Fields

### Account

The account associated with the user assigned in the `User` field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Account__c</code>	Lookup Relationship ( <a href="#">Account</a> )
Input or output with API (with any data size)	<code>account</code>	ID

### Account Accessible

Whether the stored payment is available to all users associated with the account.

Usage	Field Name	Data Type
Input on object	<code>ccrz__AccountAccessible__c</code>	Checkbox
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Account Number

Account number for the payment.

- For a purchase order (PO), this value is the entire PO number.
- For a credit card or bank account, this value is likely the account number's last four digits.

Usage	Field Name	Data Type
Input on object	ccrz__AccountNumber__c	Text (255)
Input or output with API (with any data size)	accountNumber	String

**Account Type**

The method of payment, such as purchase order (PO), credit card, and so on. For more specific payment method details, such as the specific type of credit card, use `Payment Type`.

Usage	Field Name	Data Type
Input on object	ccrz__AccountType__c	Text (255)
Input or output with API (with any data size)	accountType	String

**Address**

The billing address associated with the payment.

Usage	Field Name	Data Type
Input on object	ccrz__Address__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with any data size)	address	ID

**CC Stored Payment Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfid	String

**Display Name**

The user-friendly name that a buyer enters for the stored payment.

Usage	Field Name	Data Type
Input on object	ccrz__DisplayName__c	Text (255)
Input or output with API (with any data size)	displayName	String

**Effective Account ID**

The effective account associated with the payment.

Usage	Field Name	Data Type
Input on object	ccrz__EffectiveAccountID__c	Text (20)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Enabled**

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	ccrz__Enabled__c	Checkbox
Input or output with API (with any data size)	enabled	Boolean

**End Date**

The last day that the stored payment is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__EndDate__c	Date
Input or output with API (with any data size)	endDate	Date

**Exp Month**

The two-digit expiration month for the payment, such as for a credit card expiration date.

Usage	Field Name	Data Type
Input on object	ccrz__ExpMonth__c	Number (2, 0)
Input or output with API (with any data size)	expMonth	Integer

**Exp Year**

The four-digit expiration year for the payment, such as for a credit card expiration date.

Usage	Field Name	Data Type
Input on object	ccrz__ExpYear__c	Number (4, 0)
Input or output with API (with any data size)	expYear	Integer



**Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	ccrz__Name__c	Auto Number
Input or output with API (with any data size)	name	String

**Payment Type**

More specific payment method details for `Account Type`. For example, if `Account Type` is `Credit Card`, then this value can be the type of credit card.

Usage	Field Name	Data Type
Input on object	ccrz__PaymentType__c	Text (255)
Input or output with API (with any data size)	paymentType	String

**Read Only**

Whether a user can modify the stored payment details.

Usage	Field Name	Data Type
Input on object	ccrz__ReadOnly__c	Checkbox
Input or output with API (with any data size)	readOnly	Boolean

**Sequence**

Order of the stored payment within the list of stored payments. Lower values display first.

Usage	Field Name	Data Type
Input on object	ccrz__Sequence__c	Number (18, 0)
Input or output with API (with any data size)	sequence	Integer

**Start Date**

The first day that the stored payment is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Stored Payment ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__StoredPaymentId__c	Text (255)
Input or output with API (with any data size)	storedPaymentId	String

**Storefront**

The name of the storefront where the stored payment is available.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with any data size)	storefront	String

**Subaccount Number**

Subaccount used with Account Number.

Usage	Field Name	Data Type
Input on object	ccrz__SubaccountNumber__c	Text (255)
Input or output with API (with any data size)	subaccountNumber	String

**Token**

Tokenized form of the payment information.

Usage	Field Name	Data Type
Input on object	ccrz__Token__c	Text (255)
Input or output with API (with any data size)	token	String

**User**

The user that created the transaction payment.

Usage	Field Name	Data Type
Input on object	ccrz__User__c	Lookup Relationship ( <a href="#">User</a> )
Input or output with API (with any data size)	The API doesn't return data for this field.	

## Deprecated Fields

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Default	ccrz__Default__c	Checkbox

SEE ALSO:

[Extension Examples for Stored Payments](#)

[Salesforce Help Configure Payment Options Available on the Checkout Page](#)

[Salesforce Help Allow a Buyer to Manage Stored Payments on the My Account Page](#)

## ccrz\_\_E\_StorefrontAssociation\_\_c

Represents a link between a particular account, a particular storefront, and a particular Experience Cloud site.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiStorefront`

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

## Fields

### Account

The account for the storefront association.

Usage	Field Name	Data Type
Input on object	ccrz__Account__c	Lookup Relationship ( <a href="#">Account</a> )
Input or output with API (with any data size)	account	ID

### Account Group

The account group that corresponds to the account for the storefront association.

Usage	Field Name	Data Type
Input on object	ccrz__AccountGroup__c	Lookup Relationship ( <a href="#">ccrz__E_AccountGroup__c</a> )
Input or output with API (with any data size)	accountGroup	ID

### CC StorefrontAssociation Name

An internal-facing name for the record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	sfdcName	String

### Community Name

The name of the Experience Cloud site for the storefront association. This value is the site's name, not its URL path.

Usage	Field Name	Data Type
Input on object	ccrz__CommunityName__c	Text (255)
Input or output with API (with any data size)	communityName	String

### Default

Whether the storefront is the default for the Experience Cloud site.

Usage	Field Name	Data Type
Input on object	ccrz__Default__c	Checkbox

Usage	Field Name	Data Type
Input or output with API (with any data size)	default	Boolean

**RefId**

The external ID of the object that the `RefType` field specifies. The `RefId` field is automatically populated depending on other values set for the storefront association.

Usage	Field Name	Data Type
Input on object	ccrz__RefId__c	Text (255)
Input or output with API (with any data size)	refId	String

**RefType**

The type of object—account, account group, or Experience Cloud site—linked to the storefront association. This field is automatically populated depending on other values set for the storefront association.

Usage	Field Name	Data Type
Input on object	ccrz__RefType__c	Text (255)
Input or output with API (with any data size)	refType	String

**Storefront**

The name of the storefront linked to the storefront association.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (40)
Input or output with API (with any data size)	storefront	String

**Deprecated Fields**

The following fields exist on the object but aren't used by the managed package. Don't use these fields in any development or customization.

Field Label	Field Name	Data Type
Object ID	ccrz__Object_Id__c	Text (40)

Field Label	Field Name	Data Type
Object Type	ccrz__Object_Type__c	Text (40)

SEE ALSO:


[Salesforce Help Create Associations between Multiple Storefronts and Experience Cloud Sites](#)

## ccrz\_\_E\_SubProdTerm\_\_c

Represents subscription options that you make available for a specific product.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz\\_\\_E\\_SubProdTerm\\_\\_c](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:


[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz\_\_E\_SubProdTermI18N\_\_c

Represents subscription options that you make available for a specific product, localized for a specific storefront locale.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz\\_\\_E\\_SubProdTermI18N\\_\\_c](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz\_\_E\_Subscriber\_Page\_\_c

Represents a custom page that you add to the storefront.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** `ccrz.ccApiSubscPage`

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

## Fields

### Bridge Allowed

Whether the subscriber page is used with `CC_Bridge`. For [Create a Subscriber Page](#)

### CC Subscriber Page Name

An internal-only name for the record.

Usage	Field Name	Data Type
Input on object	Name	Text (80)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### End Date

The last day when the subscriber page is available on the storefront.

Usage	Field Name	Data Type
Input on object	<code>ccrz__EndDate__c</code>	Date
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Layout Template

The template to use for the subscriber page's body content:

- When this field is blank, the subscriber page uses the default one-column layout (`cc_tmpl_OneColRD`).
- To use a subscriber template, enter `cc_tmpl_SubscriberTemplate` for `Layout Template`, and then specify the subscriber template Visualforce page name in the `Subscriber Layout Template` field.

For usage, see [Create a Subscriber Page](#).

Usage	Field Name	Data Type
Input on object	ccrz__LayoutTemplate__c	Text (255)
Input or output with API (with any data size)	layoutTemplate	String

### Page Include

The name of the Visualforce page to use for the subscriber page.



**Tip:** Because this page is subscriber code that you've created and is outside of the managed package, prefix the page name with `c__`, such as `c__cc_example_subscriberPage`.

For usage, see [Create a Subscriber Page](#).

Usage	Field Name	Data Type
Input on object	ccrz__PageInclude__c	Text (255)
Input or output with API (with any data size)	pageInclude	String

### Page Key

A short value (three characters or fewer) for the URL query parameter that B2B Commerce references for showing your subscriber page on the storefront. For example, for a page named `cc_example_subscriberPage`, enter `xsp`.



**Note:** This value is also referenced in storefront configuration settings that apply to your subscriber page. The full programmatic path of a configuration setting—module, configuration, storefront, and page key—is limited to 40 characters. Make sure that the page key value is unique, but keep it as short as possible.

For usage, see [Create a Subscriber Page](#).

Usage	Field Name	Data Type
Input on object	ccrz__PageKey__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Start Date

The first day when the subscriber page is available on the storefront.

Usage	Field Name	Data Type
Input on object	ccrz__StartDate__c	Date
Input or output with API (with any data size)	The API doesn't return data for this field.	



**Storefront**

The name of the storefront where the subscriber page applies.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Subscriber Layout Template**

The subscriber Visualforce page name for the page's subscriber template when the value of `Layout_Template` is `cc_tmpl_SubscriberTemplate`.



**Note:** Because the page is subscriber code that you've created and is outside of the managed package, prefix the page name with `c__`, such as `c__cc_example_customTemplateName`. For more information about subscriber templates, see [Create a Subscriber Template](#).

Usage	Field Name	Data Type
Input on object	ccrz__SubscriberLayoutTemplate__c	Text (255)
Input or output with API (with any data size)	subscriberLayoutTemplate	String

**Title Label**

The text that you want to show for the page title.

Usage	Field Name	Data Type
Input on object	ccrz__TitleLabel__c	Text (255)
Input or output with API (with any data size)	titleLabel	String

**SEE ALSO:**

[Salesforce Help Create a Subscriber Page](#)

[Salesforce Help Modify the Column Layout of a Storefront Page](#)

**ccrz\_\_E\_Subscription\_\_c**

Represents a buyer's selected subscription after they place an order.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz\\_\\_E\\_Subscription\\_\\_c](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz\_\_E\_TaxRate\_\_c

Represents a tax percentage for a state in a particular country or territory. The Checkout page applies this tax rate to all taxable products, depending on the specified shipping address.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**!** **Important:** The default CC Tax Rate object and [ccrz.cc\\_hk\\_TaxCalculation](#) extension point class in the B2B Commerce API are intended for demonstration purposes only. This object and API aren't meant to be used as-is in a production org. Instead, they provide a starting point that you can extend for implementing the tax calculations that are specific to your solution. Or, integrate an external tax calculation service, such as Avalara, and install the service's add-on, if available.

## API Usage

**Class used with:** [ccrz.cc\\_hk\\_TaxCalculation](#)

**Fetch sizing:** Doesn't apply to records of this object.

## Fields

### Country Code

ISO 3166-1 Alpha 2 country or region code.

Usage	Field Name	Data Type
Input on object	ccrz__CountryCode__c	Text (128)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**State Code**

ISO 3166-2 code for a US state or Canadian province.

Usage	Field Name	Data Type
Input on object	ccrz__StateCode__c	Text (128)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Tax Rate**

The tax rate percentage for the specific US state or Canadian province.

Usage	Field Name	Data Type
Input on object	ccrz__TaxRate__c	Percent (12, 3)
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Tax Rate Auto Number**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Tax Rate ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__TaxRateId__c	Text (128)
Input or output with API (with any data size)	The API doesn't return data for this field.	

## SEE ALSO:

[Salesforce Help Show Fees and Charges in the Order Review Section of the Checkout Page](#)

**ccrz\_\_E\_Term\_\_c**

Represents terms and conditions shown to the buyer during checkout.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## API Usage

**Class used with:** [ccrz.ccApiTerms](#)

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

## Fields

### CC Term Auto Number

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	The API doesn't return data for this field.	

### Description

Explanatory text for the term, which appears below the title on the Checkout page.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Description__c</code>	Text Area (Rich) (32768)
Input or output with API (with any data size)	<code>description</code>	String

### Enabled

Whether the record is active in your org's data.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Enabled__c</code>	Checkbox

Usage	Field Name	Data Type
Input or output with API (with any data size)	The API doesn't return data for this field.	

**Locale**

The locale code for the term.

Usage	Field Name	Data Type
Input on object	ccrz__Locale__c	Text (5)
Input or output with API (with any data size)	locale	String

**Storefront**

The storefronts where the term applies.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Picklist (Multi-select)
Input or output with API (with any data size)	storefront	String

**Title**

The label or heading for the term, which displays on the Checkout page with a checkbox that a buyer must select before they can continue.

Usage	Field Name	Data Type
Input on object	ccrz__Title__c	Text (100)
Input or output with API (with any data size)	title	String

**Type**

The term type, such as Billing, Shipping, Digital, or All.

Usage	Field Name	Data Type
Input on object	ccrz__Type__c	Picklist

Usage	Field Name	Data Type
Input or output with API (with any data size)	type	String

SEE ALSO:

[Salesforce Help Set Up Terms and Conditions on the Checkout Page](#)

## ccrz\_\_E\_TransactionPayment\_\_c

Represents a payment made when a buyer completes checkout.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### API Usage

**Class used with:** [ccrz.ccApiTransactionPayment](#)

**Fetch sizing:** An API `fetch` query always evaluates the same fields for records of this object, regardless of any [specific data size request in your input map](#).

### Fields

#### Account

The account associated with the user assigned in the `User` field.

Usage	Field Name	Data Type
Input on object	ccrz__Account__c	Lookup Relationship ( <a href="#">Account</a> )
Input or output with API (with any data size)	account	ID

#### Account Number

Account number for the payment.

- For a purchase order (PO), this value is the entire PO number.
- For a credit card or bank account, this value is likely the account number's last four digits.

Usage	Field Name	Data Type
Input on object	ccrz__AccountNumber__c	Text (255)
Input or output with API (with any data size)	accountNumber	String

### Account Type

The method of payment, such as purchase order (PO), credit card, and so on. For more specific payment method details, such as the specific type of credit card, use `Payment Type`.

Usage	Field Name	Data Type
Input on object	ccrz__AccountType__c	Text (255)
Input or output with API (with any data size)	accountType	String

### Address

The billing address associated with the payment.

Usage	Field Name	Data Type
Input on object	ccrz__Address__c	Lookup Relationship ( <a href="#">ccrz__E_ContactAddr__c</a> )
Input or output with API (with any data size)	address	ID

### Amount

The transaction's actual payment amount. This value can differ from `Request Amount`.

Usage	Field Name	Data Type
Input on object	ccrz__Amount__c	Currency (18, 2)
Input or output with API (with any data size)	amount	Decimal

### Bill To

The account that is responsible for fulfilling the payment.

Usage	Field Name	Data Type
Input on object	ccrz__BillTo__c	Lookup Relationship ( <a href="#">Account</a> )

Usage	Field Name	Data Type
Input or output with API (with any data size)	billTo	ID

**CC Order**

The order record that this payment applies to.

Usage	Field Name	Data Type
Input on object	ccrz__CCOrder__c	Lookup Relationship ( <a href="#">ccrz__E_Order__c</a> )
Input or output with API (with any data size)	CCOrder	ID

**CC Subscription**

The subscription that this payment applies to.

Usage	Field Name	Data Type
Input on object	ccrz__CCSubscription__c	Lookup Relationship ( <a href="#">ccrz__E_Subscription__c</a> )
Input or output with API (with any data size)	CCSubscription	ID

**CC Transaction Payment Name**

A unique, automatically assigned ID for the record.

Usage	Field Name	Data Type
Input on object	Name	Auto Number
Input or output with API (with any data size)	sfdcName	String

**Comments**

Free text associated with the payment. Typically, comments are added when the transaction is created.

Usage	Field Name	Data Type
Input on object	ccrz__Comments__c	Text Area (Long) (32768)
Input or output with API (with any data size)	comments	String



**Contact**

The contact associated with the user assigned in the `User` field.

Usage	Field Name	Data Type
Input on object	<code>ccrz__Contact__c</code>	Lookup Relationship ( <a href="#">Contact</a> )
Input or output with API (with any data size)	<code>contact</code>	ID

**Currency ISO Code**

The ISO 4217 currency code for the payment, such as *USD* or *JPY*.

Usage	Field Name	Data Type
Input on object	<code>ccrz__CurrencyISOCode__c</code>	Text (10)
Input or output with API (with any data size)	<code>currencyISOCode</code>	String

**Expiration Month**

The two-digit expiration month for the payment, such as for a credit card expiration date.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ExpirationMonth__c</code>	Number (2, 0)
Input or output with API (with any data size)	<code>expirationMonth</code>	Integer

**Expiration Year**

The four-digit expiration year for the payment, such as for a credit card expiration date.

Usage	Field Name	Data Type
Input on object	<code>ccrz__ExpirationYear__c</code>	Number (4, 0)
Input or output with API (with any data size)	<code>expirationYear</code>	Integer

**Invoice Applied For**

The invoice that this payment fulfills.

- For a summary invoice, this invoice is the summary invoice.
- For a standard invoice, this invoice is the same record as the invoice referenced by `Invoice Applied To`.

Usage	Field Name	Data Type
Input on object	ccrz__InvoiceAppliedFor__c	Lookup Relationship (ccrz__E_Invoice__c)
Input or output with API (with any data size)	invoiceAppliedFor	ID

### Invoice Applied To

The invoice that this payment applies to.

- For a summary invoice, this invoice is the child invoice that a subpayment applies to.
- For a standard invoice, this invoice is the same record as the invoice referenced by `Invoice Applied For`.

Usage	Field Name	Data Type
Input on object	ccrz__InvoiceAppliedTo__c	Lookup Relationship (ccrz__E_Invoice__c)
Input or output with API (with any data size)	invoiceAppliedTo	ID

### Parent Transaction Payment

The transaction payment information that applies to multiple payments. When a transaction payment fulfills multiple invoices:

- A parent transaction payment can contain the information shared by all payments.
- Child transaction payments can define specific payment amounts for each individual invoice.

Usage	Field Name	Data Type
Input on object	ccrz__ParentTransactionPayment__c	Lookup Relationship (ccrz__E_TransactionPayment__c)
Input or output with API (with any data size)	parentTransactionPayment	ID
	parentTransactionPaymentR	Map<String, Object>, which contains the following fields: <ul style="list-style-type: none"> <li>• sfdcName</li> <li>• sfid</li> <li>• transactionPaymentId</li> </ul>

### Payment Type

More specific payment method details for `Account Type`. For example, if `Account Type` is `Credit Card`, then this value can be the type of credit card.

Usage	Field Name	Data Type
Input on object	ccrz__PaymentType__c	Text (255)

Usage	Field Name	Data Type
Input or output with API (with any data size)	paymentType	String

**Request Amount**

A requested payment amount. This value can differ from `Amount`.

Usage	Field Name	Data Type
Input on object	ccrz__RequestAmount__c	Currency (18, 2)
Input or output with API (with any data size)	requestAmount	Decimal

**Sold To**

The account that placed the order. This value probably matches the `Sold To` account on an invoice, or the `Ship To` account on an order.

Usage	Field Name	Data Type
Input on object	ccrz__SoldTo__c	Lookup Relationship ( <a href="#">Account</a> )
Input or output with API (with any data size)	soldTo	ID

**Source Transaction Payment**

The transaction payment information that was the source for this payment. For example, in an authorization and capture chain, the authorization is the source transaction payment and the capture is the current transaction payment.

Usage	Field Name	Data Type
Input on object	ccrz__SourceTransactionPayment__c	Lookup Relationship ( <a href="#">ccrz__E_TransactionPayment__c</a> )
Input or output with API (with any data size)	sourceTransactionPayment	ID
	sourceTransactionPaymentR	Map<String, Object>, which contains the following fields: <ul style="list-style-type: none"> <li>• <code>sfdcName</code></li> <li>• <code>sfid</code></li> <li>• <code>transactionPaymentId</code></li> </ul>

**Stored Payment**

The original `ccrz__E_StoredPayment__c` for the payment, if applicable.

Usage	Field Name	Data Type
Input on object	ccrz__StoredPayment__c	Lookup Relationship ( <a href="#">ccrz__E_StoredPayment__c</a> )
Input or output with API (with any data size)	storedPayment	ID

**Storefront**

The storefront for the payment.

Usage	Field Name	Data Type
Input on object	ccrz__Storefront__c	Text (255)
Input or output with API (with any data size)	storefront	String

**Sub Account Number**

Subaccount used with Account Number.

Usage	Field Name	Data Type
Input on object	ccrz__SubAccountNumber__c	Text (255)
Input or output with API (with any data size)	accountNumber	String

**Token**

Tokenized form of the payment information.

Usage	Field Name	Data Type
Input on object	ccrz__Token__c	Text (255)
Input or output with API (with any data size)	token	String

**Transaction Code**

Code associated with the transaction. For example, for a credit card payment, this value is the specific transaction ID.

Usage	Field Name	Data Type
Input on object	ccrz__TransactionCode__c	Text (255)
Input or output with API (with any data size)	transactionCode	String

**Transaction Payment ID**

A unique ID that associates the record with an external system.

Usage	Field Name	Data Type
Input on object	ccrz__TransactionPaymentId__c	Text (255)
Input or output with API (with any data size)	transactionPaymentId	String

**Transaction Subcode**

A secondary code associated with the transaction. For example, if the transaction failed, this value can be failure's error code.

Usage	Field Name	Data Type
Input on object	ccrz__TransactionSubcode__c	Text (255)
Input or output with API (with any data size)	transactionSubcode	String

**Transaction Timestamp**

The date and time when the transaction occurred.

Usage	Field Name	Data Type
Input on object	ccrz__TransactionTS__c	Date/Time
Input or output with API (with any data size)	transactionTS	Datetime

**Transaction Type**

Provider-specific type of transaction, such as AUTH or CAPTURE for a credit card payment.

Usage	Field Name	Data Type
Input on object	ccrz__TransactionType__c	Text (255)
Input or output with API (with any data size)	transactionType	String

**User**

The user that created the transaction payment.

Usage	Field Name	Data Type
Input on object	ccrz__User__c	Lookup Relationship ( <a href="#">User</a> )
Input or output with API (with any data size)	user	ID

**Verification Code**

An alternative to `Transaction Code`.

Usage	Field Name	Data Type
Input on object	<code>ccrz__VerificationCode__c</code>	Text (255)
Input or output with API (with any data size)	<code>verificationCode</code>	String

## SEE ALSO:

[Salesforce Help Configure Payment Options Available on the Checkout Page](#)

[Salesforce Help Generate an Invoice for a Buyer's Order](#)

## Global API Classes for B2B Commerce for Visualforce

---

B2B Commerce for Visualforce defines Apex API classes and methods that provide programmatic access to records of B2B Commerce custom objects. For example, the `ccrz.cccApiProduct` class provides methods for querying and returning data for your storefront's products.

The API classes are included in the managed package's namespace, `ccrz`. The class and method definitions use the `global` access modifier, which means they are accessible from all Apex code in your org. For example, you can write a custom Visualforce page that invokes a particular B2B Commerce for Visualforce API method call.



**Tip:** The B2B Commerce for Visualforce API isn't intended to replace high-volume data operations. Although the API can handle certain bulky operations, sometimes using direct queries or [Data Manipulation Language \(DML\)](#) can be more efficient.



**Warning:** The `revise` method available for some classes doesn't allow a guest user to update records. If you're creating subscriber code that requires a guest user to update records, you have the following options:

- If you're using B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11), invoke the `ccrz.cccLogic.updateObjects` utility method.
- Execute the DML operation in your custom Apex class using the `without sharing` accessor.

## IN THIS SECTION:

[Get Started with the B2B Commerce for Visualforce API](#)

This step-by-step tutorial shows how to construct a typical API call for B2B Commerce for Visualforce. First, define the scope of data that you want to request, and submit the request to the appropriate method. Then, parse the data that's returned.

[ccrz.cccApi](#)

Defines the base constants and exceptions that all methods of the B2B Commerce for Visualforce global API reference or return. The methods of this class check for data size flags on method calls and handle registration for custom logic and service implementations.

[ccrz.cccApiAccount](#)

Query, create, or modify account records that have B2B Commerce for Visualforce custom fields.

[ccrz.cccApiAddress](#)

Query, create, or update bill-to or ship-to address details for your storefront's contact addresses, such as those on cart and order records.

[ccrz.ccApiAddressBook](#)

Query, create, update, or delete account address book records.

[ccrz.ccApiAttribute](#)

Query attribute records defined for products when the storefront uses attribute-driven commerce.

[ccrz.ccApiCart](#)

Query, create, update, or remove cart records.

[ccrz.ccApiCategory](#)

Query category assignments for products and data for the storefront's category taxonomy.

[ccrz.ccApiConfig](#)

Query storefront configuration data, or load new JSON-formatted configuration data.

[ccrz.ccApiContact](#)

Query contact records associated with your B2B Commerce for Visualforce storefront users.

[ccrz.ccApiCoupon](#)

Query or clone coupon records, and programmatically apply or remove coupons from a shopping cart.

[ccrz.ccApiI18N](#)

Query locales and locale-specific resources for internationalizing a storefront.

[ccrz.ccApiInvoice](#)

Query invoice records.

[ccrz.ccApiMenu](#)

Query menus and menu items.

[ccrz.ccApiOrder](#)

Query, search, split, reorder, or cancel order records.

[ccrz.ccApiPriceList](#)

Query price lists and price list items, which specify an account group's entitled prices for particular products.

[ccrz.ccApiPrivateCache](#)

Query, write, or update cache records for data that must stay private to your org, such as configuration data. For data that's safe for any storefront user to see—such as page labels or the category tree—use methods of the `ccrz.ccApiPublicCache` class.

[ccrz.ccApiProduct](#)

Query product records, price entitled products, and return other entitled product details, including records related to each product, such as composite products and product media.

[ccrz.ccApiProductIndex](#)

Query, create, update, or remove product index records, which B2B Commerce for Visualforce uses for quickly populating content on the Product List page version 2.0.

[ccrz.ccApiPromotion](#)

Query promotion records.

[ccrz.ccApiPublicCache](#)

Query, write, or update cache records for data that's safe for any storefront user to see, such as page labels or the category tree. To query, write, or update cache records for data that must stay private to your org—such as configuration data—use methods of the `ccrz.ccApiPrivateCache` class.

[ccrz.ccApiRelatedProduct](#)

Query related product records, which define relationships between products, such as accessories or competitive or alternative SKUs.

[ccrz.ccApiRichContent](#)

Query rich content records used for custom storefront content.

[ccrz.ccApiSeller](#)

Query seller records by IDs or by geographic location.

[ccrz.ccApiSpec](#)

Query specification definitions and product spec index records.

[ccrz.ccApiSiteIndex](#)

Query, create, update, or remove site index records.

[ccrz.ccApiStoredPayment](#)

Query, create, update, and delete payments saved in the My Wallet section of the My Account page.

[ccrz.ccApiStorefront](#)

Query storefront association records, which link between a particular account, a particular storefront, and a particular Experience Cloud site.

[ccrz.ccApiSubProdTerm](#)

Query subscription options that are available for a specific product.

[ccrz.ccApiSubscPage](#)

Query subscriber page records, and handle the case where a specified subscriber page isn't found.

[ccrz.ccApiSubscription](#)

Query subscription records associated with placed orders.

[ccrz.ccApiTerms](#)

Query definitions of terms and conditions shown to the buyer during checkout.

[ccrz.ccApiTransactionPayment](#)

Query details of payments made when buyers complete checkout.

[ccrz.ccApiUser](#)

Query, create, or modify user records that have B2B Commerce for Visualforce custom fields.

[ccrz.ccApiWishList](#)

Query `ccrz__E_Cart__c` records where the `ccrz__CartType__c` field is `WishList`.

SEE ALSO:

[Apex Class Definition](#)

## Get Started with the B2B Commerce for Visualforce API

This step-by-step tutorial shows how to construct a typical API call for B2B Commerce for Visualforce. First, define the scope of data that you want to request, and submit the request to the appropriate method. Then, parse the data that's returned.

Say that you wanted to programmatically return data for two of your products whose IDs you know. The `ccrz.ccApiProduct` class provides a `fetch` method, which you can call from your code to get product data for a set of IDs, names, SKUs, or other criteria.

IN THIS SECTION:

1. [Request Data from the B2B Commerce for Visualforce API](#)

In this step, you construct a `Map<String, Object>` that specifies the scope of the data that you're querying.



## 2. [Process the Data Returned from the B2B Commerce for Visualforce API](#)

In this step, pass the input map you previously created to the `ccrz.ccApiProduct.fetch` method, and handle the data that the method returns.

## 3. [Versions in B2B Commerce for Visualforce API Calls](#)


Each B2B Commerce for Visualforce API call requires a `ccrz.ccAPI.API_VERSION` key in the input map. The version defines the scope of input data that the method can accept, and the specific output data that the method returns.

## 4. [Sizing and Scoping Options for a B2B Commerce for Visualforce API Method Call](#)

Depending on the complexity of your storefront data, some B2B Commerce for Visualforce API calls can return a huge volume of data. For example, the `ccrz.ccApiProduct` class returns product records, but by default executes subqueries that return product media, product specs, and other related records. Sometimes, the solution you're building requires all of that data. Other times, the solution requires only the product records, and some or none of the related records. Maybe the solution requires only a fraction of all the possible product data the API can return.

## Request Data from the B2B Commerce for Visualforce API

In this step, you construct a `Map<String, Object>` that specifies the scope of the data that you're querying.

 **Tip:** Most methods in the B2B Commerce for Visualforce API accepts a `Map<String, Object>` of input data, and returns a `Map<String, Object>` of output data.

1. Set up a `Map<String, Object>` for the data you're requesting from the API.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{ };
```

2. Specify the IDs of the product records whose data you want to query.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'}
};
```


The `ccrz.ccApiProduct.PRODUCTIDLIST` key is defined as an acceptable input for the `ccrz.ccApiProduct.fetch` method. Although we haven't invoked the `fetch` method yet, we are setting up data that the method accepts. The `ccrz.ccApiProduct.PRODUCTIDLIST` key requires a *set* of product IDs.

3. Specify the version of the API call by adding the `ccrz.ccApi.API_VERSION` key to the `productFetchQuery` map.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'},

    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION
};
```

The `ccrz.ccApi.API_VERSION` key is defined on the `ccrz.ccApi` base class. This key is *required* for every B2B Commerce for Visualforce API call, but [the specific version you provide can affect how your call requests and returns data](#) on page 288.

 **Tip:** We recommend that new implementations always use `ccrz.ccApi.CURRENT_VERSION` whenever possible. This pointer ensures that your subscriber code always references the most recent version of the API, even when you upgrade to a new major release.

4. Add a `ccrz.ccApi.SIZING` key to the `productFetchQuery` map.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'},
```

```

    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> { }
};

```

The `ccrz.ccApi.SIZING` key is defined on the `ccrz.ccApi` base class. This key specifies how much data the API call returns. Because this key is also a `Map<String, Object>`, you can [add multiple parameters to return the exact scope of data that you want](#). Some methods automatically trigger subqueries to other API calls for querying records from multiple objects. You can use one `ccrz.ccApi.SIZING` key to control how much data is returned for both the primary query—such as product data—and any subqueries.

- To restrict how much product data the call returns, add a `ccrz.ccApiProduct.ENTITYNAME` key to the `ccrz.ccApi.SIZING` map.

```

Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'},

    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> { }
    }
};

```

- To specify how many fields you want to return for each product record, add the `ccrz.ccApi.SZ_DATA` key to `ccrz.ccApiProduct.ENTITYNAME`.

```

Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'},

    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
        }
    }
};

```

The `ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S` flag tells the API to return only the [small set of data on each product record](#). To return more fields, specify a [higher size value](#). For some methods, certain data size values also determine which related data your query returns.



**Note:** When you call the B2B Commerce for Visualforce API from a Visualforce page or Apex remote action, B2B Commerce has already constructed the context for the call: the storefront and its configuration, the current user, which effective account is selected, and so on. This information lives in the `ccrz.cc_CallContext` utility class. But outside of a Visualforce page or Apex remote action, B2B Commerce can't construct this context automatically. If your subscriber code is calling the API from outside of a Visualforce page or Apex remote action, your code must set this context. For more information about call context, see the `ccrz.cc_CallContext` object.

Next, add code that invokes `ccrz.ccApiProduct.fetch` and handles the data that the method returns.

## Process the Data Returned from the B2B Commerce for Visualforce API

In this step, pass the input map you previously created to the `ccrz.ccApiProduct.fetch` method, and handle the data that the method returns.

1. Add `try` and `catch` statements.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'},

    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
        }
    }
};

try {
    // Pass the input data
} catch (Exception e) {
    // Error handling...
}
```

2. In the `try` statement, pass the input map to the `ccrz.ccApiProduct.fetch` method and capture it in a new map, `productFetchData`.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'},

    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
        }
    }
};

try {
    Map<String, Object> productFetchData = ccApiProduct.fetch(productFetchQuery);
} catch (Exception e) {
    // Error handling...
}
```

The `ccrz.ccApiProduct.fetch` method returns a list of product records, hashed under the `ccrz.ccApiProduct.PRODUCTLIST` key of the `productFetchData` output map. The `ccrz.ccApiProduct.PRODUCTLIST` key is a `List<Map<String, Object>>` collection, where each product record is a `Map<String, Object>`.

3. To process the returned list, create a `List<Map<String, Object>>` collection to capture the content of the `ccrz.ccApiProduct.PRODUCTLIST` key. Cast the hashed product data returned in the `ccrz.ccApiProduct.PRODUCTLIST` key of the `productFetchData` map to the list.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'},

    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
        }
    }
};
```

```

    }
  }
};

try {
  Map<String, Object> productFetchData = ccApiProduct.fetch(productFetchQuery);
  if (productFetchData.get(ccApiProduct.PRODUCTLIST) != null) {
    List<Map<String, Object>> myProductDataList = (List<Map<String, Object>>)
productFetchData.get(ccApiProduct.PRODUCTLIST);
  }
} catch (Exception e) {
  // Error handling...
}

```

4. With the product data in a list, you can use Apex list parsing to extract data. For example, to get a product's name, create a string and cast the value of product's `sfdcName` field to the string.

```

Map<String, Object> productFetchQuery = new Map<String, Object>{
  ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
  ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'},

  ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
      ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
    }
  }
};

try {
  Map<String, Object> productFetchData = ccrz.ccApiProduct.fetch(productFetchQuery);
  if (productFetchData.get(ccrz.ccApiProduct.PRODUCTLIST) != null) {
    List<Map<String, Object>> myProductDataList = (List<Map<String, Object>>)
productFetchData.get(ccrz.ccApiProduct.PRODUCTLIST);
    String productName = (String) myProductDataList[0].get('sfdcName');
  }
} catch (Exception e) {
  // Error handling...
}

```

## Versions in B2B Commerce for Visualforce API Calls

Each B2B Commerce for Visualforce API call requires a `ccrz.ccAPI.API_VERSION` key in the input map. The version defines the scope of input data that the method can accept, and the specific output data that the method returns.

Usually, you can simplify your implementation by pointing to whichever version is most recent in your version of the managed package:

```
ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
```



**Tip:** We recommend that new implementations always use `ccrz.ccApi.CURRENT_VERSION` whenever possible. This pointer ensures that your subscriber code always references the most recent version of the API, even when you upgrade to a new major release.

If necessary, reference a specific API version if you depend on a certain scope of data or functionality.

```
ccrz.ccApi.API_VERSION => 9
```

For example, if your subscriber code depends on functionality in an older version, pointing to the latest version can potentially break your subscriber code.

Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21 (version 4.13)	12
B2B Commerce for Visualforce Spring '20 (version 4.12)	11
B2B Commerce for Visualforce Summer '19 (version 4.11)	10
B2B Commerce for Visualforce Spring '19 (version 4.10)	9
B2B Commerce for Visualforce Summer '18 (version 4.9)	8
CloudCraze 4.8	7
CloudCraze 4.7	6
CloudCraze 4.6	5
CloudCraze 4.5	4
CloudCraze 4.4	3
CloudCraze 4.3	2
CloudCraze 4.2 and earlier	1

When you reference a specific API version and upgrade the managed package, you can continue to reference the older API version. But, as a best practice, we recommend that you update all your calls to reference the latest available version when possible.

If you try to call a version that doesn't exist, the method returns an exception:

- `ccrz.BelowMinAPIVersionException` if you try to call a version less than 1
- `ccrz.ExceedsMaxAPIVersionException` if you try to call a version greater than the latest available version


Even though you can still call an older version, don't mix versions when you call more than one method within the same operation. For example, if your code calls both `ccrz.ccApiCart.addTo` and `ccrz.ccApiCart.price`, specify the same version in both calls.

```
ccrz.ccApiCart.addTo(new Map<String, Object>{
    ccrz.ccApi.API_VERSION => 9
});

ccrz.ccApiCart.price(new Map<String, Object>{
    ccrz.ccApi.API_VERSION => 9
});
```

## Sizing and Scoping Options for a B2B Commerce for Visualforce API Method Call

Depending on the complexity of your storefront data, some B2B Commerce for Visualforce API calls can return a huge volume of data. For example, the `ccrz.ccApiProduct` class returns product records, but by default executes subqueries that return product media, product specs, and other related records. Sometimes, the solution you're building requires all of that data. Other times, the solution requires only the product records, and some or none of the related records. Maybe the solution requires only a fraction of all the possible product data the API can return.

 **Warning:** If you extend any default data service provider classes, such as `ccrz.ccServiceProduct`, avoid modifying any size or scope options already specified. Otherwise, you risk breaking default functionality that calls the data service provider classes that you've extended. We recommend that you use these size and scope options only in subscriber code that calls the B2B Commerce for Visualforce global APIs.


To scope the size of a method's query, first include a `ccrz.ccApi.SIZING` key in your input map.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> { }
};
```

Inside the `ccrz.ccApi.SIZING` key, include one or more `ENTITYNAME` keys for the class whose data you're querying.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> { }
    }
};
```

Inside `ENTITYNAME`, include one or more of the available sizing flags.

 **Note:** Some classes have methods that query different types of records, and require differentiated `ENTITYNAME` keys for the different record types. For example, the `ccrz.ccApiPriceList` class has methods that fetch price list records and a method that fetches price list item records. A key named `ccrz.ccApiPriceList.ENTITYNAME` defines the size of price list records, but doesn't apply to price list item records. For sizing price list items, use `ccrz.ccApiPriceList.ENTITYNAME_PLI`. To make sure you're using the correct key for your data, check the reference for the method you're calling.

### ccrz.ccApi.SZ\_DATA

Scope the number of fields returned for the primary object that the method queries. For example, restrict the number of fields on a product record that a `ccrz.ccApiProduct` method returns. This key accepts t-shirt sizes for values:

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
        }
    }
};
```

The specific fields returned for each size vary depending on the method. In general, the following conventions apply:

Value	Fields Returned
<code>ccrz.ccApi.SZ_XS</code>	Only <code>sfdcName</code> and <code>sfid</code> . For child-level objects, this size also includes the ID of the parent object. For example, calling this size on a price list item also includes the ID of the parent price list.
<code>ccrz.ccApi.SZ_S</code>	All fields from <code>ccrz.ccApi.SZ_XS</code> , plus information such as storefront or SKU.
<code>ccrz.ccApi.SZ_M</code>	All fields from <code>ccrz.ccApi.SZ_S</code> , plus information such as status, type, and sequence.
<code>ccrz.ccApi.SZ_L</code> (default size if none is specified)	All fields from <code>ccrz.ccApi.SZ_M</code> , plus information such as long description, price totals, and adjustments.
<code>ccrz.ccApi.SZ_XL</code>	All possible fields.

For some methods, certain data size values also trigger a *subquery* for data from objects that are related to the object of the primary query. For example, specifying `SZ_M` for `ccrz.ccApiProduct.fetch` results in an `EProductMedias` field in the return data. This field isn't specified on the product object, but provides data from the product media records related to each product returned in the primary query. For better control of subquery data, use the `ccrz.ccApi.SZ_REL` key on page 291.

## ccrz.ccApi.SZ\_REL

Specify the exact types of related records you want to return as a subquery, regardless of which subqueries your specified `ccrz.ccApi.SZ_DATA` value triggers by default

For example, passing `SZ_M` to `ccrz.ccApiProduct.fetch` returns product media and subscriptions records related to the product. What if you want all the product fields that `SZ_M` returns, but none of the subscriptions data? And what if you want to also return product specs records, which by default is available only for `SZ_L`?

Use the `ccrz.ccApi.SZ_REL` flag and pass a `List<String>` with the names of specific subqueries you want to return:

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
            ccrz.ccApi.SZ_REL => new List<String>{'E_ProductMedias__r', 'Product_Specs__r'}
        }
    }
};
```



**Tip:** For information about specific subqueries available for a particular primary object, see the reference for that object.

The content of `ccrz.ccApi.SZ_REL` takes precedence over any subqueries returned by default for your `ccrz.ccApi.SZ_DATA` value.

If you want to exclude subqueries entirely, pass an empty list:

```
ccrz.ccApi.SZ_REL => List<String>{ }
```

## ccrz.ccApi.SZ\_ASSC

In addition to subqueries, many B2B Commerce for Visualforce methods chain calls to methods of other classes for querying other relevant object data. For example, the `ccrz.ccApiCart.fetch` method returns cart data, which references addresses and products for a cart. To return address and product data, the `ccrz.ccApiCart.fetch` method also calls `ccrz.ccApiAddress.fetch` and `ccrz.ccApiProduct.fetch`, respectively. You can optionally disable calls to other APIs for the current call.

```
Map<String, Object> cartFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
            ccrz.ccApi.SZ_ASSC => false
        }
    }
};
```


You can exclude both subqueries and other object data queries:

```
Map<String, Object> cartFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
            ccrz.ccApi.SZ_ASSC => false,
            ccrz.ccApi.SZ_REL => new List<String>{ }
        }
    }
};
```

## ccrz.ccApi.SZ\_SKIPTRZ

Most B2B Commerce for Visualforce API methods return data as a `List<Map<String, Object>>`, where each `Map<String, Object>` in the list contains hashed object data. If you don't want to work with hashed object data, you can return a list of `sObjects` instead.

```
Map<String, Object> cartFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_SKIPTRZ => true
        }
    }
};
```

 **Note:** The B2B Commerce for Visualforce API serializes the `sObject` data into separate strings instead of a consolidated `sObject`. Use this flag when you prefer to avoid the extra processing time required to serialize the data.

## ccrz.ccApi.SZ\_REFETCH

Some methods in the B2B Commerce for Visualforce API—such as `create` or `revise` methods—support an implicit refetch operation for fetching updated records after the method's logic executes. For example, several `ccrz.ccApiCart` methods implicitly refetch cart records after modifying their content so that the updated returns are returned immediately within the same transaction.



You can flag the API call to perform this refetch operation, which can help you optimize your subscriber code by avoiding redundant queries.

```
Map<String, Object> myCartData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
            ccrz.ccApi.SZ_ASSC => false,
            ccrz.ccApi.SZ_REFETCH => true
        }
    }
};
```

For information about whether a specific method supports the refetch flag, see the reference for that method.

### ccrz.ccApi.SZ\_SKIPFLTCK

Optionally, you can bypass any keys that scope the size of a method call.

```
ccrz.ccApi.SZ_SKIPFLTCK => true
```

**!** **Important:** In general, avoid open-ended queries and specify the scope and size of your method call. Setting this flag can create an unconstrained `WHERE` clause, which can significantly affect your org's resources. Use this flag sparingly.

### Combine Multiple Flags

You can include more than one class's `ENTITYNAME` in your input map's `ccrz.ccApi.SIZING` key, and you can include more than one sizing flag on each `ENTITYNAME`.

```
Map<String, Object> myRequestData = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XL
        },
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
            ccrz.ccApi.SZ_REL => new List<String>{'ccrz__E_ProductMedias__r',
            'ccrz__Product_Specs__r'}
        }
    }
};
```

The sizing request applies to all subsequent calls by any API class in the same thread. For example, maybe you limit the size of product data on a `ccrz.ccApiCart` call but then also query that product data with maximum sizing. In this case, only the first sizing specification is respected.

## ccrz.ccApi

Defines the base constants and exceptions that all methods of the B2B Commerce for Visualforce global API reference or return. The methods of this class check for data size flags on method calls and handle registration for custom logic and service implementations.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Globally Available Keys

This class defines the following global keys. In the input map for a specific method call, specify values for these keys to establish the scope and size of your query. A method's output map also uses these keys for return data.

### `ccrz.ccApi.API_VERSION`

String that defines the [version of a method call](#).

### `ccrz.ccApi.CACHE_DATA`

Map<String, Object> that specifies cached data for a particular object, such as the category tree or page labels.

### `ccrz.ccApi.CACHE_KEYMAP`

Map<String, Object> that uniquely identifies the requested cache data and ensures that repeated calls always return the same set of data. The `ccrz.ccLogic.prepareCacheKey` method returns the value for this key.

### `ccrz.ccApi.CACHE_INP`

Map<String, Object> that contains input data for the cache query.

### `ccrz.ccApi.CACHE_KEY`


String that specifies a key for the cached data. The `ccrz.ccLogic.prepareCacheKey` method returns the value for this key.

### `ccrz.ccApi.CACHE_TTL`

Integer that specifies the number of seconds for the time to live (TTL) for a particular type of cached data, or how long that data remains in the cache. For example, a TTL value of `300` is 300 seconds, or five minutes. When the TTL expires, the data is no longer available in the cache.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApi.MESSAGES**

List<[ccrz.cc\\_bean\\_Message](#)>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

**ccrz.ccApi.API\_SIZING**

Map<[String](#), [Object](#)> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

The `ccrz.ccApi` class also defines the following constants that can be used as [subkeys for the SIZING map](#).

- `SZ_DATA`
- `SZ_REL`
- `SZ_ASSC`

## Methods

### IN THIS SECTION:

#### [checkSizing](#)

Checks for any specified `ccrz.ccApi.SIZING` key and sets the `ccrz.ccApi.SZ_DIR` variable for sizing subsequent API calls within the same API transaction.

#### [checkVersion](#)

Checks whether the input map includes the `ccrz.ccApi.API_VERSION` key.

#### [flushCache](#)

Clears the B2B Commerce for Visualforce partition of your org's Platform Cache, which invalidates any existing org-level cache data. Use this method when the B2B Commerce data in Platform Cache is updated from some external source, such as a batch job. It's always safe to flush the cache.

#### [registerService](#)

Registers a new data service class in the B2B Commerce for Visualforce API, such as for add-on packages or custom APIs that reference existing B2B Commerce data services. This method is rarely used in B2B Commerce solutions.

#### [registerLogic](#)

Registers a new logic service class in the B2B Commerce for Visualforce API, such as for add-on packages or custom APIs that reference existing B2B Commerce logic services. This method is rarely used in B2B Commerce solutions.

#### [unregisterService](#)

Removes a data service registration from the B2B Commerce for Visualforce API. This method is typically used only during development, and is rarely used in production.

#### [unregisterLogic](#)

Removes a logic service registration from the B2B Commerce for Visualforce API. This method is typically used only during development, and is rarely used in production.

## ccrz.ccApi.checkSizing

Checks for any specified `ccrz.ccApi.SIZING` key and sets the `ccrz.ccApi.SZ_DIR` variable for sizing subsequent API calls within the same API transaction.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> checkSizing (Map<String, Object>)
```

## ccrz.ccApi.checkVersion

Checks whether the input map includes the `ccrz.ccApi.API_VERSION` key.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Integer checkSizing (Map<String, Object>)
```

## Inputs (Required)

Integer that specifies the version of the B2B Commerce for Visualforce API.

## Outputs

`Map<String, Object>` that can include the following exceptions for an API call:

**BelowMinAPIVersionException**

Returned when the `ccrz.ccApi.API_VERSION` key specifies a version that is less than `1`.

**ExceedsMaxAPIVersionException**

Returned when the `ccrz.ccApi.API_VERSION` key specifies a version that exceeds the latest version of the API that's available.

**NoApiVersionException**

Returned when the input map doesn't include the `ccrz.ccApi.API_VERSION` key.

## ccrz.ccApi.flushCache

Clears the B2B Commerce for Visualforce partition of your org's Platform Cache, which invalidates any existing org-level cache data. Use this method when the B2B Commerce data in Platform Cache is updated from some external source, such as a batch job. It's always safe to flush the cache.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global static Map<String, Object> flushCache (Map<String, Object>)
```

### Inputs (Required)

`Map<String, Object>`, which isn't used for any operations.

### Outputs

`Map<String, Object>` that's always empty.

SEE ALSO:

[Salesforce Help Improve Storefront Performance with Platform Cache](#)

## ccrz.ccApi.registerService

Registers a new data service class in the B2B Commerce for Visualforce API, such as for add-on packages or custom APIs that reference existing B2B Commerce data services. This method is rarely used in B2B Commerce solutions.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> registerService (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>`, where each key is the key that the B2B Commerce for Visualforce API references for managing the data service. The value for each key is the namespaced default class of the data service.

For example, to register `myDataService` and `anotherDataService`, set up your input map with the following keys:

```
ccrz.ccApi.registerService(new Map<String, Object>{
    'myDataService' => 'namespace.myDataService'
    'anotherDataService' => 'namespace.anotherDataService'
});
```



Tip:

- The combination of the key name and default class name can't exceed 40 characters.
- Don't use *S* or *L* as the first character of the key name. Those characters are reserved for internal use.

## Outputs

`Map<String, Object>` that includes all registered data service and logic service providers.

## ccrz.ccApi.registerLogic

Registers a new logic service class in the B2B Commerce for Visualforce API, such as for add-on packages or custom APIs that reference existing B2B Commerce logic services. This method is rarely used in B2B Commerce solutions.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> registerLogic (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>`, where each key is the key that the B2B Commerce for Visualforce API references for managing the logic service. The value for each key is the namespaced default class of the logic service.

For example, to register `myLogicService` and `anotherLogicService`, set up your input map with the following keys:

```
ccrz.ccApi.registerLogic (new Map<String, Object>{
    'myLogicService' => 'namespace.myLogicService'
    'anotherLogicService' => 'namespace.anotherLogicService'
});
```

### Tip:

- The combination of the key name and default class name can't exceed 40 characters.
- Don't use `S` or `L` as the first character of the key name. Those characters are reserved for internal use.

## Outputs

`Map<String, Object>` that includes all registered data service and logic service providers.

## ccrz.ccApi.unregisterService

Removes a data service registration from the B2B Commerce for Visualforce API. This method is typically used only during development, and is rarely used in production.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> unregisterService (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>`, where each key is the key used to register the data service. The value for each key is a Boolean:

Value	Usage
<code>true</code>	The method first checks for an override of the data service. If an override exists, the method returns <code>ccrz.ccApi.UnregisterServiceException</code> .
<code>false</code>	The method first checks for an override of the data service. If an override exists, the override is removed. If an override doesn't exist, the data service is unregistered.

## Outputs

`Map<String, Object>` that includes all registered data service and logic service providers.

## ccrz.ccApi.unregisterLogic

Removes a logic service registration from the B2B Commerce for Visualforce API. This method is typically used only during development, and is rarely used in production.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> unregisterLogic (Map<String, Object>)
```



## Inputs (Required)

`Map<String, Object>`, where each key is the key used to register the logic service. The value for each key is a Boolean.

Value	Usage
<code>true</code>	The method first checks for an override of the logic service. If an override exists, the method returns <code>ccrz.ccApi.UnregisterServiceException</code> .
<code>false</code>	The method first checks for an override of the logic service. If an override exists, the override is removed. If an override doesn't exist, the logic service is unregistered.

## Outputs

`Map<String, Object>` that includes all registered data service and logic service providers.

## ccrz.ccApiAccount

Query, create, or modify account records that have B2B Commerce for Visualforce custom fields.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

#### [createAccount](#)

Creates an account record from the fields and values that you specify.

#### [fetch](#)

Executes a SOQL query of account records and returns a list of accounts that match your query.

#### [getAnonymous](#)

Returns the account associated with the storefront guest user.

#### [getCurrent](#)

Returns the account associated with the user currently logged in to the storefront.

[revise](#)

Updates fields on existing account records.

[validateNew](#)

Checks new account records for missing fields and returns a list of errors.

## SEE ALSO:

[Account \(B2B Commerce for Visualforce\)](#)


## ccrz.ccApiAccount.createAccount

Creates an account record from the fields and values that you specify.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> createAccount (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider


[ccrz.ccLogicAccountCreateAccount](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.


#### `ccrz.ccApiAccount.ACCOUNT_NEW`

List<String> that specifies fields and values for creating the [account record](#). Only certain account fields are writable when creating the record:

Field	Required?	Type
billingAddress	Required	Map<String, Object> (see the following table)
companyName	Optional	String
customField1	Optional	String
customField2	Optional	String
firstName	Required	String
lastName	Required	String
primaryPhone	Required	String
shippingAddress	Required	Map<String, Object> (see the following table)
taxExemptFlag	Required	Boolean
userName	Required	String

The `billingAddress` and `shippingAddress` maps can contain the following writable fields:

Field	Required?	Type
address1	Required	String
address2	Optional	String
city	Required	String
country	Required	String
countryCode	Optional	String
postalCode	Required	String
state	Required	String
stateCode	Optional	String

 **Tip:** For more information about how these fields map to account object fields, see the reference for [ccrz.ccLogicAccountCreateAccount.processAccount](#).

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiAccount.fetch` for returning the created accounts.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiAccount.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs


Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### **ccrz.ccApiAccount.ACCOUNTS**

List<Map<String, Object>> that contains only one Map<String, Object>, which represents the created [account record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

### **Example: Example**

Create an account by passing in a JSON-formatted definition of a new account record (`newAccountJSON`, not shown). Set the `ccrz.ccApi.SZ_REFETCH => true` flag on the input map and return the created account in the list of accounts.

```
Map<String, Object> newAccountMap =
    (Map<String, Object>) JSON.deserializeUntyped(newAccountJSON);

Map<String, Object> inputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiAccount.ACCOUNT_NEW => newAccountMap,
    ccrz.ccApi.SIZING => new Map<String, Object>{
        ccrz.ccApiAccount.ENTITYNAME => new Map<String, Object>{
```

```

        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}

try {
    Map<String, Object> outputData = ccrz.ccApiAccount.createAccount(inputData);
    if(outputData.get(ccrz.ccApiAccount.ACCOUNTS) != null){
        // THIS IS THE REFETCHED ACCOUNT
        // The cast to List<Map<String, Object>> is necessary...
        List<Map<String, Object>> accounts = (List<Map<String, Object>>)
outputData.get(ccrz.ccApiAccount.ACCOUNTS);
        // DO SOMETHING WITH THE DATA.
    }

} catch (Exception e) {
    // Error handling...
}

```

**SEE ALSO:**

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

## ccrz.ccApiAccount.fetch

Executes a SOQL query of account records and returns a list of accounts that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceAddress`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

#### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

#### `ccrz.ccApiAccount.ID`

String of a specific account ID to query.

#### `ccrz.ccApiAccount.IDS`

`List<String>` of specific account IDs to query:

```
new List<String>{'Account_ID_1', 'Account_ID_2'}
```

#### `ccrz.ccApiAccount.NAME`

String that specifies the name of an account to query.

## Outputs


`Map<String, Object>` that can include the following keys:

#### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

#### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiAccount.ACCOUNTS**

List<Map<String, Object>>, where each Map<String, Object> represents an [account record](#).



**Tip:** To specify which fields the API returns for each account record, [request a specific data size](#) for the `ccrz.ccApiAccount.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAccount.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields are available with different data sizes, see the [account](#) reference.

## SEE ALSO:

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

**ccrz.ccApiAccount.getAnonymous**

Returns the account associated with the storefront guest user.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

**Signature**

```
global static Map<String, Object> getAnonymous (Map<String, Object>)
```

**Service Layer Classes****Logic Service Provider**


[ccrz.ccLogicAccountGetAnonymous](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`


Integer that indicates which API version was used for the query.

### `ccrz.ccApi.MESSAGES`

`List<ccrz.cc_bean_Message>`, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiAccount.ACCOUNTS`

`List<Map<String, Object>>` that contains only one `Map<String, Object>`, which represents the [account](#) associated with the storefront guest user.

 **Note:** This method chains a call to `ccrz.ccApiAccount.fetch` for returning this key.

### Example: Example

Parse the `ccrz.ccApiAccount.ACCOUNTS` return key and get the only included `Map<String, Object>`, which describes the storefront guest user account. Capture the account name in a string variable.

```
Map<String, Object> inputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
}

try {
    Map<String, Object> outputData = ccrz.ccApiAccount.getAnonymous(inputData);
    if (outputData.get(ccrz.ccApiAccount.ACCOUNTS) != null) {
        // The cast to List<Map<String, Object>> is necessary...
```



```

        List<Map<String, Object>> accounts = (List<Map<String, Object>>)
outputData.get(ccrz.ccApiAccount.ACCOUNTS);

        // Get the Current Account Account
        Map<String, Object> anonymousAccount = accounts.get(0);

        // The cast, again, is necessary...
        String anonymousAccountName = (String)anonymousAccount.get('sfdcName');
    }
} catch (Exception e) {
    // Error handling...
}

```

**SEE ALSO:**

[Salesforce Help: Allow a B2B Commerce for Visualforce Guest Buyer to Browse Your Storefront](#)


## ccrz.ccApiAccount.getCurrent

Returns the account associated with the user currently logged in to the storefront.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> getCurrent(Map<String, Object>)
```

### Service Layer Classes

**Logic Service Provider**

[ccrz.ccLogicAccountGetCurrent](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`


Integer that indicates which API version was used for the query.

### `ccrz.ccApi.MESSAGES`

`List<ccrz.cc_bean_Message>`, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiAccount.ACCOUNTS`

`List<Map<String, Object>>` that contains only one `Map<String, Object>`, which represents the [account](#) associated with the current storefront user.

 **Note:** This method chains a call to `ccrz.ccApiAccount.fetch` for returning this key.

### Example: Example

Parse the `ccrz.ccApiAccount.ACCOUNTS` return key and get the only included `Map<String, Object>`, which describes the current storefront user's account. Capture the account name in a string variable.

```
Map<String, Object> inputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
}

try {
    Map<String, Object> outputData = ccrz.ccApiAccount.getCurrent(inputData);
    if (outputData.get(ccrz.ccApiAccount.ACCOUNTS) != null) {
        // The cast to List<Map<String, Object>> is necessary...
```

```

        List<Map<String, Object>> accounts = (List<Map<String, Object>>)
outputData.get(ccrz.ccApiAccount.ACCOUNTS);

        // Get the Current Account Account
        Map<String, Object> currentAccount = accounts.get(0);

        // The cast, again, is necessary...
        String currentAccountName = (String)currentAccount.get('sfdcName');
    }
} catch (Exception e) {
    // Error handling...
}

```

**SEE ALSO:**

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)


## ccrz.ccApiAccount.revise

Updates fields on existing account records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

 **Warning:** This method doesn't allow a guest user to update records. If you're creating subscriber code that requires a guest user to update records, you have the following options:

- If you're using B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11), invoke the [ccrz.ccLogic.updateObjects utility method](#).
- Execute the DML operation in your custom Apex class using the [without sharing accessor](#).

### Signature

```
global static Map<String, Object> revise (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceAccount`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### `ccrz.ccApiAccount.ACCOUNTS`

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the fields and values to update for an [account record](#).



**Important:** You can update the following fields on an account record, but you must include the `sfid` for each record you're updating.

Field	Type	Corresponding Account Field
<code>billingCity</code>	String	BillingAddress
<code>billingCountry</code>	String	
<code>billingPostalCode</code>	String	
<code>billingState</code>	String	
<code>billingStreet</code>	String	
<code>EAAccountGroup</code>	ID	<code>ccrz__AccountGroup__c</code>
<code>parentId</code>	String	ParentId
<code>phone</code>	Boolean	Phone
<code>preventAddrBookSave</code>	Boolean	<code>ccrz__PreventAddrBookSave__c</code>
<code>preventNewAddrCheckout</code>	Boolean	<code>ccrz__PreventNewAddrCheckout__c</code>
<code>preventNewBillAddrCO</code>	String	<code>ccrz__PreventNewBillAddrCO__c</code>
<code>sfdcName</code>	String	Name
<code>sfId</code>	String	SalesforceId
<code>shippingCity</code>	String	ShippingAddress
<code>shippingCountry</code>	String	
<code>shippingPostalCode</code>	String	
<code>shippingState</code>	String	

Field	Type	Corresponding Account Field
shippingStreet	String	
sourceSystem	String	ccrz__SourceSystem__c
taxExemptAccount	Boolean	ccrz__TaxExemptAccount__c

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiAccount.fetch` for returning the updated accounts.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiAccount.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs

Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

### **ccrz.ccApi.MESSAGES**

List<ccrz.cc\_bean\_Message>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.


### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### **ccrz.ccApiAccount.ACCOUNTS**

List<Map<String, Object>>, where each Map<String, Object> represents an updated [account record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

SEE ALSO:

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)


## ccrz.ccApiAccount.validateNew

Checks new account records for missing fields and returns a list of errors.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> validateNew (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

```
ccrz.ccLogicAccountValidateNew
```

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### **ccrz.ccApiAccount.ACCOUNT\_NEW**

`Map<String, Object>` of fields and values to validate for a new [account record](#). The following fields are required in the map:

- firstName
- lastName
- primaryPhone
- language
- currencyCode
- username
- companyName
- companyType
- billingAddress
  - address1
  - city
  - postalCode
- shippingAddress
  - address1
  - city
  - postalCode

## Outputs

Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

### **ccrz.ccApi.MESSAGES**

List<ccrz.cc\_bean\_Message>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

### Example: Example

Validate the fields in JSON-formatted definition of a new account record (`newAccountJSON`, not shown). Capture the error messages and process them.

```
Map<String, Object> newAccountMap =
(Map<String, Object>) JSON.deserializeUntyped(newCustomerJSON);

Map<String, Object> inputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiAccount.ACCOUNT_NEW => newAccountMap
}

try {
    List<ccrz.cc_bean_Message> errorMessages =
(List<ccrz.cc_bean_Message>) ccrz.ccApiAccount.validateNew(inputData).get(ccrz.ccApiAccount.MESSAGES)

    // Assume there are errors.
    for (ccrz.cc_bean_Message msg : errorMessages) {
        // Get the error message...
        castMsg.labelId;
    }
} catch (Exception e) {
    // Exception handling...
}
```

SEE ALSO:

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

## ccrz.ccApiAddress

Query, create, or update bill-to or ship-to address details for your storefront's contact addresses, such as those on cart and order records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



## Methods

### IN THIS SECTION:

#### [createAddress](#)

Creates a contact address record from the fields and values that you specify. On the storefront, this method is invoked by default when a buyer enters new address information during line-level independence (LLI) checkout flow.

#### [fetch](#)

Executes a SOQL query of contact address records and returns a list of contact addresses that match your query.

#### [revise](#)

Updates fields on existing contact address records.

### SEE ALSO:

#### [ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)


## ccrz.ccApiAddress.createAddress

Creates a contact address record from the fields and values that you specify. On the storefront, this method is invoked by default when a buyer enters new address information during line-level independence (LLI) checkout flow.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> createAddress (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


[ccrz.ccLogicAddressCreateAddress](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:


### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiAddress.ADDRESSLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the fields and values to use for creating a [contact address record](#).

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiAddress.fetch` for returning the created addresses.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

### `ccrz.ccApi.SKIP_OWNERID`

Boolean

Value	Usage
<code>true</code>	Set the Owner field on the new sObject to a default value.
<code>false</code> (default)	Set the Owner field on the new sObject to the value of <code>ccrz.cc_CallContext.currUserId</code> .

## Outputs

`Map<String, Object>` that can include the following keys:


### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`


Boolean

Value	Usage
<code>true</code>	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiAddress.ADDRESSLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a created [contact address record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

**ccrz.ccApiAddress.ADDRESSIDLIST**

List<String> of Salesforce IDs of the created contact address records.

## SEE ALSO:

[Salesforce Help Account Address Books and Contact Addresses for B2B Commerce for Visualforce Buyers](#)

[Salesforce Help Standard Checkout versus Line-Level Independence \(LLI\) Checkout](#)

[Salesforce Help Specify Whether a Buyer Can Modify Addresses on the Checkout Page](#)


## ccrz.ccApiAddress.fetch

Executes a SOQL query of contact address records and returns a list of contact addresses that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceAddress`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

#### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

#### `ccrz.ccApiAddress.ADDRESSIDLIST`

`Set<String>` of specific contact address IDs to query:

```
new Set<String>{'Contact_Address_ID_1', 'Contact_Address_ID_2'}
```

## Outputs


`Map<String, Object>` that can include the following keys:

#### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.


#### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

#### `ccrz.ccApiAddress.ADDRESSLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c`.

 **Tip:** To specify which fields the API returns for each contact address record, [request a specific data size](#) for the `ccrz.ccApiAddress.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields are available with different data sizes, see the `ccrz__E_ContactAddr__c` reference.

### Example: Example

Query a pair of contact address IDs, and capture the first contact's first name in a string variable.

```
Set<String> addressIdList = new Set<String>{'ADDRESS_ID_1', 'ADDRESS_ID_2'};

Map<String, Object> addressFetchQuery = new Map<String, Object>{
    ccrz.ccApiAddress.ADDRESSIDLIST => addressIdList,
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
};

try {
    Map<String, Object> addressFetchData = ccrz.ccApiAddress.fetch(addressFetchQuery);

    if (addressFetchData.get(ccrz.ccApiAddress.ADDRESSLIST) != null) {
        List<Map<String, Object>> outputAddressList = (List<Map<String, Object>>)
addressFetchData.get(ccrz.ccApiAddress.ADDRESSLIST);
        String firstName = (String) outputAddressList[0].get('firstName');
    }
} catch (Exception e) {
    // Error handling...
}
```



## ccrz.ccApiAddress.revise

Updates fields on existing contact address records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

-  **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.
-  **Warning:** This method doesn't allow a guest user to update records. If you're creating subscriber code that requires a guest user to update records, you have the following options:
  - If you're using B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11), invoke the [ccrz.ccLogic.updateObjects utility method](#).
  - Execute the DML operation in your custom Apex class using the [without sharing accessor](#).

## Signature

```
global static Map<String, Object> revise (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

ccrz.ccServiceAddress

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

-  **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes [ccrz.ccApiAddress.fetch](#) for returning the updated addresses.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

### **ccrz.ccApiAddress.ADDRESSLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the fields and values to update for a [contact address record](#).

-  **Important:** You can update any fields on a contact address record, but you must include the `sfiid` for each record you're updating.

## Outputs


Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

### **ccrz.ccApiAddress.ADDRESSLIST**

List<Map<String, Object>>, where each Map<String, Object> represents an updated [contact address record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.


## ccrz.ccApiAddressBook

Query, create, update, or delete account address book records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Tip:** The methods of this class work with the `ccrz__E_AccountAddressBook__c` object, and not `ccrz__E_AddressBook__c`, which is deprecated.

## Methods

### IN THIS SECTION:

#### [createAddressBook](#)

Creates an account address book record from the fields and value that you specify.

#### [fetch](#)

Executes a SOQL query of account address book records and returns a list of account address books that match your query.

#### [removeAddressBook](#)

Deletes account address book records whose IDs you specify.

#### [revise](#)

Updates fields on existing account address book records.

### SEE ALSO:

[ccrz\\_\\_E\\_AccountAddressBook\\_\\_c](#)

## ccrz.ccApiAddressBook.createAddressBook

Creates an account address book record from the fields and value that you specify.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> createAddressBook (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

[ccrz.ccLogicAddressBookCreateAddressBook](#)



## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:


### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiAddressBook.ADDRESSBOOKS`

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the fields and values to use for creating an [account address book record](#).

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that can include a flag to include the created address book records with the return data. This flag ensures that the `ccrz.ccApiAddressBook.fetch` method also executes within this method's logic chain.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiAddressBook.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

For more information, see [Sizing and Scoping Options for a B2B Commerce for Visualforce API Method Call](#).

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.


### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiAddressBook.ADDRESSBOOKS**

List<Map<String, Object>>, where each Map<String, Object> represents a created [account address book record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

**Example: Example**

Create a default billing address.

```
Map<String, Object> addressBookToInsert = new Map<String, Object>{
    'addressType' => 'Billing',
    'default' => 'true',
    'typeReadOnly' => 'false',
    'EContactAddress' => 'some_Contact_Address_ID',
    'accountId' => 'some_Account_ID'
};

Map<String, Object> createAddressBookInput = (new Map<String, Object>{
    ccrz.ccApiAddressBook.ADDRESSBOOKS => new
List<Map<String, Object>>{addressBookToInsert},
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiAddressBook.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_REFETCH => true
        }
    }
});

try {
    Map<String, Object> createdAddressBookData =
ccrz.ccApiAddressBook.createAddressBook(createAddressBookInput);
    if (createdAddressBookData.get(ccrz.ccApiAddressBook.ADDRESSBOOKS) != null) {
        List<Map<String, Object>> addressBookList = (List<Map<String,
Object>>)createdAddressBookData.get(ccrz.ccApiAddressBook.ADDRESSBOOKS);
        Map<String, Object> addressBook = addressBookList.get(0);
        String addressType = (String)addressBook.get('addressType');
    }
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[Salesforce Help Account Address Books and Contact Addresses for B2B Commerce for Visualforce Buyers](#)


**ccrz.ccApiAddressBook.fetch**

Executes a SOQL query of account address book records and returns a list of account address books that match your query.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceAddressBook`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#).

 **Note:** Specific data size requests don't apply to this method. The same fields are always returned for each object, regardless of any specific data size request in your input map.

### `ccrz.ccApiAddressBook.ACCOUNTID`

String that specifies an account ID whose associated address books you want to query.

### `ccrz.ccApiAddressBook.ADDRESSID`

String that specifies a [contact address ID](#) whose associated account address books you want to query.

### `ccrz.ccApiAddressBook.ISDEFAULT`

Boolean

Value	Usage
<code>true</code>	Return only the account address books where the Default field is checked.
<code>false</code> (default)	Ignore the Default field when querying account address books.

**ccrz.ccApiAddressBook.OWNERID**

String that specifies a user ID. Return only the account address books where the Owner field matches the specified user ID.

**ccrz.ccApiAddressBook.TYPE**

String that specifies the type of account address books to query, either Shipping or Billing.

**ccrz.ccApiAddressBook.ID**

String that specifies an account address book ID to query.

**ccrz.ccApiAddressBook.IDS**

List<String> of specific account address book IDs to query:

```
new List<String>{'Address_Book_ID_1', 'Address_Book_ID_2'}
```

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiAddressBook.ADDRESSBOOKS**

List<Map<String, Object>>, where each Map<String, Object> represents an [account address book record](#).



**Note:** Each address book record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.

**ccrz.ccApiAddress.ADDRESSLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [contact address record](#) that's related to an account address book.

### Example: Example

Query the account address books related to a specific account, and capture the account address book type in a string variable.

```
Map<String, Object> addressBookFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiAddressBook.ACCOUNTID => 'Some_Account_ID'
};

try {
    Map<String, Object> addressBookFetchData =
ccrz.ccApiAddressBook.fetch(addressBookFetchQuery);
    if (addressBookFetchData.get(ccrz.ccApiAddressBook.ADDRESSBOOKS) != null) {
        List<Map<String, Object>> addressBookList = (List<Map<String,
Object>>)addressBookFetchData.get(ccrz.ccApiAddressBook.ADDRESSBOOKS);
        Map<String, Object> addressBook = addressBookList.get(0);
        String addressType = (String)addressBook.get('addressType');
    }
} catch (Exception e) {
    // Error handling...
}
```


## ccrz.ccApiAddressBook.removeAddressBook

Deletes account address book records whose IDs you specify.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> removeAddressBook(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicAddressBookRemoveAddressBook`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### `ccrz.ccApiAddressBook.IDS`

`List<String>` of Salesforce IDs of the [account address book records](#) to delete.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

### Outputs


`Map<String, Object>` that can include the following keys:

#### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

#### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .



### Example: Example

Delete a specific account address book.

```
Map<String, Object> removeAddressBookInput = (new Map<String, Object>){
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiAddressBook.IDS => new List<String>{ 'some_Account_Address_Book_ID' }
});

try {
    Map<String, Object> removeAddressBookResult =
ccrz.ccApiAddressBook.removeAddressBook(removeAddressBookInput);
} catch (Exception e) {
```

```
// Error handling...
}
```



## ccrz.ccApiAddressBook.revise

Updates fields on existing account address book records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

-  **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.
-  **Warning:** This method doesn't allow a guest user to update records. If you're creating subscriber code that requires a guest user to update records, you have the following options:
  - If you're using B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11), invoke the [ccrz.ccLogic.updateObjects utility method](#).
  - Execute the DML operation in your custom Apex class using the [without sharing accessor](#).

### Signature

```
global static Map<String, Object> revise(Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider

ccrz.ccServiceAddressBook

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`


`Map<String, Object>` that can include a flag to include the updated address book records with the return data. This flag ensures that the `ccrz.ccApiAddressBook.fetch` method also executes within this method's logic chain.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiAddressBook.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

For more information, see [Sizing and Scoping Options for a B2B Commerce for Visualforce API Method Call](#).

### `ccrz.ccApiAddressBook.ADDRESSBOOKS`

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the fields and values to update for an [account address book record](#).

 **Important:** You can pass any fields on an account address book record, but you must include the `sfid` for each record you're updating.

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.


### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiAddressBook.ADDRESSBOOKS`

`List<Map<String, Object>>`, where each `Map<String, Object>` represents an updated [account address book record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.



### Example: Example

Change the `typeReadOnly` field value to `TRUE` for a specific account address book record.

```
Map<String, Object> addressBookToUpdate = new Map<String, Object>{
    'sfid' => 'some_Account_Address_Book_ID',
    'typeReadOnly' => TRUE
};

Map<String, Object> reviseAddressBookInput = (new Map<String, Object>{
    ccrz.ccApiAddressBook.ADDRESSBOOKS => new
List<Map<String, Object>>{addressBookToUpdate},
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiAddressBook.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_REFETCH => true
        }
    }
});

try {
    Map<String, Object> reviseAddressBookResult =
ccrz.ccApiAddressBook.revise(reviseAddressBookInput);
} catch (Exception e) {
    // Error handling...
}
```

## ccrz.ccApiAttribute

Query attribute records defined for products when the storefront uses attribute-driven commerce.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

[fetch](#)

Executes a SOQL query of attribute records.

### SEE ALSO:

[Salesforce Help Set Up Attribute-Based Products](#)


## ccrz.ccApiAttribute.fetch

Executes a SOQL query of attribute records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceAttribute`

### Logic Service Provider

`ccrz.ccLogicAttrFetch`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

**Inputs (Optional)**

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

**ccrz.ccApiAttribute.ATTRIBUTEIDS**

`Map<String, Object>` where each key is a parent attribute ID and each value is a `List<String>` of child attribute IDs:

```
new Map<String, Object>{
    'ParentAttributeID1' => new List<String>{'ChildAttributeID1', 'ChildAttributeID2',
    'ChildAttribute3'},
    'ParentAttributeID2' => new List<String>{'ChildAttributeID4', 'ChildAttributeID6',
    'ChildAttribute5'}
}
```

**ccrz.ccApiAttribute.BYAID**

Boolean

Value	Usage
<code>true</code>	Key returned attribute records by the <code>E_Attribute__c</code> field instead of ID: <pre>Map&lt;String, Object&gt;{     'ParentAttributeID1'+ccrz.ccApiPriceList.DELIMITER+'ChildAttributeID1'     =&gt; {E_Attribute__c},     'ParentAttributeID1'+ccrz.ccApiPriceList.DELIMITER+'ChildAttributeID2'     =&gt; {E_Attribute__c},     'ParentAttributeID1' =&gt; {E_Attribute__c} }</pre>
<code>false</code> (default)	Key returned attribute records by ID.

**ccrz.ccApiAttribute.IDS**

`Set<String>` of specific attribute IDs to query:

```
new Set<String>{'Attribute_ID_1', 'Attribute_ID_2'}
```


**ccrz.ccApiAttribute.LOCALE**

String that specifies one of your storefront's [supported locales](#).

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccApiAttribute.PARENTIDS**

Set<String> of parent attribute ID Platform IDs whose child attributes you want to query.

 **Important:** When you include this key in your input map, the parent attribute records aren't included in the returned attributes. To include parent IDs, specify their IDs in the `ccrz.ccApiAttribute.IDS` input key.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiPriceList.ATTRIBUTES**

A Map<String, Map<String, Object>> where each Map<String, Object> represents a return attribute, keyed by its ID. To key by `E_Attribute__c` field instead, include `ccrz.ccApiAttribute.BY_AID` in the input map.

## ccrz.ccApiCart

Query, create, update, or remove cart records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

#### [addTo](#)

Adds products to an existing cart, or creates a cart from blank input data.

#### [cartAdjustment](#)

Calls the `ccrz.cc_api_PriceAdjustment` extension point class and updates the cart and cart item prices. This method doesn't invoke any other pricing calls.

#### [cloneCart](#)

Creates a copy of an existing cart or wish list. For a cart, this method updates the `ccrz__ActiveCart__c` field on both the original cart record and its clone.

#### [create](#)

Creates a cart record. You can specify field values for the cart, or set up an empty cart with default field values.

#### [createCartItemGroup](#)

Creates a shipping group of cart items for line-level independence (LLI) checkout.

#### [createWishList](#)

Creates a wish list and wish list items based on a specific cart ID and its cart items.

#### [fetch](#)

Executes a SOQL query of cart records, and returns a list of carts that match your query.

#### [fetchCartItems](#)

Executes a SOQL query of cart item records, and returns a list of cart items that match your query.

#### [getActive](#)

Calls the `ccrz.ccApiCart.fetch` method and returns the active cart for the current account from the set of queried carts. This method sets other carts that the account can access to inactive.

#### [getDeliveryDates](#)

Calls the `ccrz.cc_api_DeliveryDate` extension point class and returns the available delivery dates for one or more shipping addresses.

#### [getShippingOptions](#)

Calls the `ccrz.cc_api_ShippingAndHandling` extension point class and returns the available shipping options for one or more shipping addresses.

#### [initItemGroups](#)

Initializes shipping groups on a specific cart for line-level independence (LLI) checkout. This method creates a single default shipping group with all cart items and the account's default shipping address. If the cart already contains at least one group, this method assigns any cart items that aren't assigned to a group to that group.

#### [place](#)

Converts a cart to a corresponding order synchronously.

#### [placeAsync](#)

Converts a cart to a corresponding order using asynchronous Apex jobs. This method is called recursively for processing multiple, small batches of cart items. When this method is first called, it first executes synchronous logic, and then enqueues recursive calls that execute asynchronous logic. Only the synchronous logic provides return data.

#### [price](#)

Reprices all the items in a cart.

### [priceAsync](#)

Reprices all the items in a cart using asynchronous Apex jobs. This method supports pricing a cart with a much larger number of items than the `ccrz.ccApiCart.price` method. This method also includes both a synchronous portion that updates the cart's status and instantiates the asynchronous reprice jobs. However, the asynchronous behavior means that there's no guarantee for how long it takes for the jobs to complete.

### [processItemGroups](#)

Moves cart items between shipping groups for line-level independence (LLI) checkout.

### [removeCart](#)

Deletes cart records, including wish lists. If a deleted cart or wish list was active for the current account, this method also determines which cart or wish list becomes active next.

### [removeCartItemGroup](#)

Deletes shipping groups of cart items for line-level independence (LLI) checkout.

### [removeFrom](#)

Removes items from a cart. This method doesn't invoke pricing logic for repricing the cart after the items have been removed.

### [repair](#)

Returns a cart where the status remains Repricing even after a repricing job completes, updates the status to Open, and resets the Repriced Date timestamp. A cart can remain in this status when a repricing job is unsuccessful or no longer available in the org. This method calls the `ccrz.ccApiCart.fetch` method for returning the affected cart.

### [revise](#)

Updates certain fields on a cart.

### [reviseCartItemGroup](#)

Updates certain fields on a cart item group record for line-level independence (LLI) checkout.

### [saveCart](#)

Calls the `ccrz.ccApiCart.validate` method for setting the cart's `ccrz__ValidationStatus__c` field. If you're using effective accounts, this method also invokes the `ccrz.cc_hk_EffectiveAccount.saveCart` method for updating cart address information.

### [setActive](#)

Sets a single cart as the active cart for a particular owner, and updates all the owner's other open carts to inactive. Finally, this method fetches the new active cart.

### [transferCart](#)

Transfers ownership of a cart to another user. By default, B2B Commerce for Visualforce calls this method when a guest user begins adding items to a cart and then registers for a storefront account. This method transfers ownership of the cart from the default owner to the newly registered user.

### [validate](#)

Provides an extension point for implementing business rules that ensure that a buyer is allowed to check out from the Shopping Cart page. By default, this method calls `ccrz.ccApiCart.fetch` and returns both the fetched cart data and a Boolean flag that indicates whether the buyer can check out.


## **ccrz.ccApiCart.addTo**

Adds products to an existing cart, or creates a cart from blank input data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

The current version of this method supports adding the following product types and pricing configurations to a cart:

- All standard [product types](#)
- Subscription products, kits, bundles, and dynamic kits based on the `ccrz__E_SubProdTerm__c` object
- [Products based on attribute-driven commerce](#)
- [Externally priced products](#)

## Signature

```
global static Map<String, Object> addTo (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccLogicCartAddTo`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

 **Note:** This method creates a cart—instead of adding products to an existing cart—when any of the following conditions are true:

- The current user isn't entitled to access the cart that the `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID` key specifies.
- The method's input map doesn't specify `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID`.

 **Note:** In version 9 or later of the API, if the specified cart is inactive when this method executes, the cart becomes active when this method completes. In version 8 or earlier of the API, the cart remains inactive when this method completes.

#### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the updated cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

#### **ccrz.ccApiCart.CART\_ID**


String that specifies the Salesforce ID of an existing cart record to add products to.

#### **ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of an existing cart record to add products to.


#### **ccrz.ccApiCart.COUPON\_CODE**

String that specifies a coupon code to apply to the cart.

 **Note:** If the input map doesn't specify either `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID`, the coupon code isn't evaluated.

#### **ccrz.ccApiCart.ISREPRICE**

Boolean

Value	Usage
<code>true</code>	Call the <code>ccrz.ccApiCart.price</code> method to reprice the cart after the items are added.  <b>Note:</b> This operation can be expensive, depending on the complexity of your cart data, but ensures that prices are current and entitlement is correct.
<code>false</code> (default)	Don't reprice cart items.

#### **ccrz.ccApiCart.ISSKIPPRICING**

Boolean

Value	Usage
<code>true</code>	Call the <code>ccrz.ccLogicProductPricing</code> class for pricing the products you're adding to the cart.
<code>false</code> (default)	Ignore pricing information for the products you're adding to the cart.



**ccrz.ccApiCart.LINE\_DATA**

List<Map<String, Object>>, where each Map<String, Object> describes a [ccrz\\_\\_E\\_CartItem\\_\\_c](#) record to create. For information about this list's keys and requirements, see the [Inputs for Cart Line Data](#) section.



**Tip:** This key also accepts a List<ccrz.ccApiCart.LineData> representation, which was required in version 5 or earlier of this method. We recommend that new solutions always use the List<Map<String, Object>> representation, which provides better compatibility with other classes of the API.

## Inputs for Cart Line Data

The input map key, `ccrz.ccApiCart.LINE_DATA`, accepts a List<Map<String, Object>>, where each <Map<String, Object>> describes a `ccrz__E_CartItem__c` to create. Each map can contain the following fields:

**ccrz.ccApiCart.LINE\_DATA\_ATTRIBUTE\_LIST**

List<ID> for an attribute-based product to add to the cart. The first ID in the list is a primary attribute, the second ID is a secondary attribute, and third ID is a tertiary attribute.

**ccrz.ccApiCart.LINE\_DATA\_EXT\_NAME**

String that specifies a value for the `ccrz__ExtName__c` field on the created `ccrz__E_CartItem__c` record. Use this key when you also specify `ccrz.ccApiCart.LINE_DATA_PRICE`.

**ccrz.ccApiCart.LINE\_DATA\_EXT\_SKU**

String that specifies a value for the `ccrz__ExtSKU__c` field on the created `ccrz__E_CartItem__c` record. Use this key when you also specify `ccrz.ccApiCart.LINE_DATA_PRICE`.

**ccrz.ccApiCart.LINE\_DATA\_LABEL**

String that specifies a value to set for the `Item Label` field on the created `ccrz__E_CartItem__c` record.



**Note:** If you're adding multiple dynamic kits to the cart at one time, we recommend that you always provide a value for this label. Otherwise, errors can occur when attempting to add dynamic kits to the cart.

**ccrz.ccApiCart.LINE\_DATA\_MINOR\_LINE\_DATAS**

List<Map<String, Object>>, where each <Map<String, Object>> contains these same fields for a component product of a composite product.

**ccrz.ccApiCart.LINE\_DATA\_PARENT\_PRODUCT\_SFID**

String that specifies the ID of a composite product. Specify this key when you're adding an aggregated product. Use `ccrz.ccApiCart.LINE_DATA_PRODUCT_SFID` for the component product (the individual style) and `ccrz.ccApiCart.LINE_DATA_PARENT_PRODUCT_SFID` for the aggregated product (parent). For more information, see [Create an Aggregated Product](#).

**ccrz.ccApiCart.LINE\_DATA\_PRICE**

Decimal that specifies the product's externally defined price. The added product inherits the properties of an externally priced product.



**Important:** Use this key only for externally priced product use cases.

**ccrz.ccApiCart.LINE\_DATA\_PRODUCT\_SFID**

Salesforce ID for the product to add to the cart.



**Note:** Specify `ccrz.ccApiCart.LINE_DATA_SKU` or `ccrz.ccApiCart.LINE_DATA_PRODUCT_SFID`, but not both.

**ccrz.ccApiCart.LINE\_DATA\_QUANTITY**


Integer that specifies the quantity for the product to add to the cart.

**ccrz.ccApiCart.LINE\_DATA\_SELLER\_SFID**

Salesforce ID of a seller, which can determine pricing and entitlement when creating the `ccrz__E_CartItem__c` record.

**ccrz.ccApiCart.LINE\_DATA\_SKU**

String that specifies the SKU for the product to add to the cart.

 **Note:** Specify `ccrz.ccApiCart.LINE_DATA_SKU` or `ccrz.ccApiCart.LINE_DATA_PRODUCT_SFID`, but not both.

**ccrz.ccApiCart.LINE\_DATA\_SUBPRODTERM\_SFID**

Salesforce ID of the `ccrz__E_SubProdTerm__c` to add with the product.

**ccrz.ccApiCart.LINE\_DATA\_SUBTOTAL**

Decimal that specifies a unique subtotal price for the quantity of externally priced products. This value overrides the default subtotal calculation (price multiplied by quantity). For example, if you specify `ccrz.ccApiCart.LINE_DATA_PRICE` of `20.00` and a `ccrz.ccApiCart.LINE_DATA_QUANTITY` of 10, the default subtotal calculation is `200.00`. With this key, you manually specify that the subtotal is some other value, such as `150.00`.

## Outputs

Map<`String`, `Object`> that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.MESSAGES**

List<`ccrz.cc_bean_Message`>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to which items were added.

 **Tip:** If this key is empty or missing from the return data, an error likely occurred. Consider rolling back your operation.

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<`String`, `Object`>>, where each Map<`String`, `Object`> represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 **Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.

- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

**ccrz.ccApiCart.INCREERROR**

Map<String, Decimal> of SKUs and their respective quantities that the method doesn't add to the cart because the `ccrz.ccApiCart.LINE_DATA_QUANTITY` input key specified quantities that violate product quantity rules.

- 📌 **Note:** You can instead specify that the method changes an invalid quantity to the next valid increment, and return a warning. For more information, see [ccrz.cc\\_api\\_ProductQuantityRule](#) and [Enforce Quantity Rules When a Buyer Adds Products to a Cart or Wish List](#).

**ccrz.ccApiCart.INCRWARN**

Map<String, Decimal> of SKUs and their respective quantities that the method changes, based on product quantity rules, from what the `ccrz.ccApiCart.LINE_DATA_QUANTITY` input key specified. The method changes the specified quantity to the next valid quantity increment, and returns a `ccrz.ccApiCart.InvalidQuantityException`.

- 📌 **Note:** You can instead specify that the method doesn't add a product with an invalid quantity. For more information, see [ccrz.cc\\_api\\_ProductQuantityRule](#) and [Enforce Quantity Rules When a Buyer Adds Products to a Cart or Wish List](#).

**ccrz.ccApiCart.INELIGIBLEPRODUCTS**

List<String> of SKUs or product IDs that the method couldn't add to the cart, such as:

- Composite products without the appropriate component products
- Products for which the method couldn't determine a price

**ccrz.ccApiCart.NONSTDPRODUCTS**

List<String> of SKUs or product IDs that the method evaluates as *non-standard*, such as:

- The composite product record for an aggregated product, but no selected component product
- Products for which the method couldn't determine a price
- Products not enabled for a particular seller

**ccrz.ccApiCart.PROSADDEDTOCART**

List<String> of SKUs or product IDs that the method added to the cart.

## Associated API Calls

This method triggers calls to the following methods of other classes:

- [ccrz.ccApiAddress.fetch](#) for returning the cart's address information.
- [ccrz.ccApiAttribute.fetch](#), if this method adds attribute-based products
- [ccrz.ccApiProduct.fetch](#) for returning the product records that correspond to the cart's items.

### 👁 Example: Example: Create a Cart from Empty Input Data

```
Boolean wasSuccessful = false;
String newCartID = null;

Map<String, Object> cartToCreate = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
};

try {
    Map<String, Object> createdCart = ccrz.ccApiCart.addTo(cartToCreate);
```

```

wasSuccessful = (Boolean)createdCart.get(ccrz.ccApi.SUCCESS);
newCartID = (String)createdCart.get(ccrz.ccApiCart.CART_ENCID);
if(String.isBlank(newCartID)){
    wasSuccessful = false;
    // Consider adding more error checking for the new cart
} catch (Exception e) {
    // Error handling...
}

```

### Example: Example: Add a SKU to an Existing Cart

```

Boolean wasSuccessful = false;
Map<String, Object> newSKUToAdd = new Map<String, Object> {
    ccrz.ccApiCart.LINE_DATA_SKU => 'productSKU',
    ccrz.ccApiCart.LINE_DATA_QUANTITY => 5
};

Map<String, Object> cartItemsToAdd = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.CART_ID => 'validCartID',
    ccrz.ccApiCart.LINE_DATA => new List<Map<String, Object>> {newSKUToAdd}
};

try {
    Map<String, Object> addToCartResults = ccrz.ccApiCart.addTo(cartItemsToAdd);
    wasSuccessful = (Boolean)addToCartResults.get(ccrz.ccApi.SUCCESS);
} catch (Exception e) {
    // Error handling...
}

```

### Example: Example: Add a SKU and Apply a Coupon Code

To reprice the cart after applying the coupon code, set `ccrz.ccApiCart.ISREPRICE => true`.

```

Boolean wasSuccessful = false;
Map<String, Object> newSKUToAdd = new Map<String, Object> {
    ccrz.ccApiCart.LINE_DATA_SKU => 'productSKU',
    ccrz.ccApiCart.LINE_DATA_QUANTITY => 5
};

Map<String, Object> cartItemsToAdd = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.CART_ID => 'validCartID',
    ccrz.ccApiCart.LINE_DATA => new List<Map<String, Object>> {newSKUToAdd},
    ccrz.ccApiCart.COUPON_CODE => 'couponCode',
    ccrz.ccApiCart.ISREPRICE => true
};

try {
    Map<String, Object> addToCartResults = ccrz.ccApiCart.addTo(cartItemsToAdd);
    wasSuccessful = (Boolean)addToCartResults.get(ccrz.ccApi.SUCCESS);
} catch (Exception e) {
    // Error handling...
}

```

 **Example: Example: Add a Dynamic Kit**

```

Boolean wasSuccessful = false;

// Dynamic kit component products
Map<String, Object> dynamicKitComponentProduct1 = new Map<String, Object> {
    ccrz.ccApiCart.LINE_DATA_SKU => 'componentSKU1',
    ccrz.ccApiCart.LINE_DATA_QUANTITY => 5
};

Map<String, Object> dynamicKitComponentProduct2 = new Map<String, Object> {
    ccrz.ccApiCart.LINE_DATA_SKU => 'componentSKU2',
    ccrz.ccApiCart.LINE_DATA_QUANTITY => 3
};

Map<String, Object> dynamicKitComponentProduct3 = new Map<String, Object> {
    ccrz.ccApiCart.LINE_DATA_SKU => 'componentSKU3',
    ccrz.ccApiCart.LINE_DATA_QUANTITY => 10
};

// Dynamic kit composite product
Map<String, Object> dynamicKitCompositeProduct = new Map<String, Object> {
    ccrz.ccApiCart.LINE_DATA_SKU => 'dynamicKitSKU',
    ccrz.ccApiCart.LINE_DATA_QUANTITY => 1,
    ccrz.ccApiCart.LINE_DATA_LABEL => 'Label for the dynamic kit',
    ccrz.ccApiCart.LINE_DATA_MINOR_LINE_DATAS => new List<Map<String, Object>> {
        dynamicKitComponentProduct1, dynamicKitComponentProduct2,
dynamicKitComponentProduct3
    }
};

Map<String, Object> cartItemsToAdd = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.CART_ID => 'validCartID',
    ccrz.ccApiCart.LINE_DATA => new List<Map<String, Object>>
{dynamicKitCompositeProduct}
};

try {
    Map<String, Object> addToCartResults = ccrz.ccApiCart.addTo(cartItemsToAdd);
    wasSuccessful = (Boolean)addToCartResults.get(ccrz.ccApi.SUCCESS);
} catch (Exception e) {
    // Error handling...
}

```

 **Example: Example: Add a Subscription**

```

Boolean wasSuccessful = false;
Map<String, Object> newSubscriptionToAdd = new Map<String, Object> {
    ccrz.ccApiCart.LINE_DATA_PRODUCT_SFID => 'productID',
    ccrz.ccApiCart.LINE_DATA_QUANTITY => 1,
    ccrz.ccApiCart.LINE_DATA_SUBPRODTERM_SFID => 'subprodtermID'
};

```

```

Map<String, Object> cartItemsToAdd = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.CART_ID => 'validCartID',
    ccrz.ccApiCart.LINE_DATA => new List<Map<String, Object>> {newSubscriptionToAdd}
};

try {
    Map<String, Object> addToCartResults = ccrz.ccApiCart.addTo(cartItemsToAdd);
    wasSuccessful = (Boolean)addToCartResults.get(ccrz.ccApi.SUCCESS);
} catch (Exception e) {
    // Error handling...
}

```

### Example: Example: Add a SKU with External Pricing

```

Boolean wasSuccessful = false;
Map<String, Object> newExternalSKUToAdd = new Map<String, Object> {
    ccrz.ccApiCart.LINE_DATA_PRODUCT_SFID => 'productID',
    ccrz.ccApiCart.LINE_DATA_EXT_NAME => 'Externally Defined Product Name'
    ccrz.ccApiCart.LINE_DATA_EXT_SKU => 'externallyDefinedProductSKU',
    ccrz.ccApiCart.LINE_DATA_PRICE => 10.00,
    ccrz.ccApiCart.LINE_DATA_QUANTITY => 5
};

Map<String, Object> cartItemsToAdd = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.CART_ID => 'validCartID',
    ccrz.ccApiCart.LINE_DATA => new List<Map<String, Object>> {newExternalSKUToAdd}
};

try {
    Map<String, Object> addToCartResults = ccrz.ccApiCart.addTo(cartItemsToAdd);
    wasSuccessful = (Boolean)addToCartResults.get(ccrz.ccApi.SUCCESS);
} catch (Exception e) {
    // Error handling...
}

```

#### SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[Salesforce Help Offer Coupons for Discounts on Storefront Purchases](#)

[Salesforce Help Types of Products for B2B Commerce for Visualforce](#)

[Salesforce Help Set Up Attribute-Based Products](#)

[Salesforce Help Set Product Prices in the Cart from an External Source](#)


## ccrz.ccApiCart.cartAdjustment

Calls the `ccrz.cc_api_PriceAdjustment` extension point class and updates the cart and cart item prices. This method doesn't invoke any other pricing calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

This class is compatible only with the following storefront page versions:

- Shopping Cart page version 1.0. Version 2.0 was introduced in [B2B Commerce for Visualforce Spring '19 \(version 4.10\)](#), but version 1.0 remains available for compatibility.
- Version 1.0 of the Order Review section of the Checkout page. Version 2.0 was introduced in [B2B Commerce for Visualforce Summer '19 \(version 4.11\)](#), but version 1.0 remains available for compatibility.

When you use version 2.0 of the Shopping Cart page or Order Review section of the Checkout page, extend the [ccrz.ccLogicCartPrice](#) logic service provider class instead.

## Signature

```
global static Map<String, Object> cartAdjustment (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccLogicCartAdjustment`, which invokes [ccrz.cc\\_api\\_PriceAdjustment](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart record to evaluate.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Outputs

Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

### **ccrz.ccApi.MESSAGES**

List<ccrz.cc\_bean\_Message>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .



### Example: Example

Return error messages from `ccrz.cc_api_PriceAdjustment` and show the messages on the storefront.

```
Boolean wasSuccessful = false;

try {
    Map<String, Object> cartAdjustmentData = ccrz.ccApiCart.cartAdjustment(new
Map<String, Object> {
        ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
        ccrz.ccApiCart.CART_ENCID => 'cart_encrypted_ID'
    });
    wasSuccessful = (Boolean).get(ccrz.ccApi.SUCCESS);
    List<ccrz.cc_bean_Message> errorMessages =
(List<ccrz.cc_bean_Message>) data.get(ccrz.ccApi.MESSAGES);
    if(!errorMessages.isEmpty()) {
        // send error messages to the front-end
    } catch (Exception e) {
        // Error handling...
    }
}
```

SEE ALSO:

[ccrz.cc\\_api\\_PriceAdjustment](#)

## ccrz.ccApiCart.cloneCart

Creates a copy of an existing cart or wish list. For a cart, this method updates the `ccrz__ActiveCart__c` field on both the original cart record and its clone.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> cloneCart (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicCartClone`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include one of the following keys. If none of these keys are included, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the original cart to clone.

### `ccrz.ccApiCart.CART_IDLIST`

`Set<String>` that can include the Salesforce ID of only one original cart to clone.



**Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the original cart to clone.

### `ccrz.ccApiCart.CART_ENCIDLIST`

`Set<String>` that can include the encrypted ID of only one original cart to clone.



**Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

 **Note:** This method checks for these keys in the following order of precedence:

1. `ccrz.ccApiCart.CART_ID`
2. `ccrz.ccApiCart.CART_ENCID`
3. `ccrz.ccApiCart.CART_IDLIST`
4. `ccrz.ccApiCart.CART_ENCIDLIST`

## Inputs (Optional)

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the cloned cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`


Integer that indicates which API version was used for the query.

### `ccrz.ccApi.MESSAGES`

`List<ccrz.cc_bean_Message>`, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cloned cart that this method creates.

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the cloned cart that this method creates.

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, which contains one Map<String, Object> that represents the clone `ccrz__E_Cart__c` record, which the `ccrz.ccApiCart.fetch` method returned.

 **Note:**

- By default, this method clones only one cart record at one time. Even though this key specifies a list, the list contains only one cart.
- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see `ccrz.ccApiCart.fetch`.

 **Note:**

- If the original cart was active, the clone becomes the new active cart, and the original becomes inactive.
- This method doesn't change the active field on a wish list. The Wish List Selector widget on the storefront sets the active field on a wish list.

 **Example: Example**

Clone a specified cart, and fetch the clone and include it in the return data.

```
Boolean wasSuccessful = false;
Map<String, Object> cartToClone = new Map<String, Object>{
    ccrz.ccApiCart.CART_ID => 'CART_ID',
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_REFETCH => true
        }
    }
};

try {
    Map<String, Object> cloneCartResult = ccrz.ccApiCart.cloneCart(cartToClone);
    wasSuccessful = (Boolean)cloneCartResult.get(ccrz.ccApi.SUCCESS);
    if (cloneCartResult.get(ccrz.ccApiCart.CART_OBJLIST) != null) {
        List<Map<String, Object>> cloneCartList = (List<Map<String, Object>>)
cloneCartResult.get(ccrz.ccApiCart.CART_OBJLIST);
        // Parse and evaluate the fetched clone
    }
} catch (Exception e) {
    // Error handling...
}
```

## SEE ALSO:

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)


**ccrz.ccApiCart.create**

Creates a cart record. You can specify field values for the cart, or set up an empty cart with default field values.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> create (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicCartCreate`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the created cart.


```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

```
}
}
```

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the `ccrz__E_Cart__c` record to create. This map can specify subkeys for field values to set on the cart record. For a list of fields, see the `ccrz__E_Cart__c` reference.

 **Note:** By default, this method creates only one cart record at one time. Even though this key specifies a list, the method evaluates only the first item in the list.

 **Note:** If you don't specify `ccrz.ccApiCart.CART_OBJLIST`, this method creates a cart with default or empty values.

## Outputs

Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.MESSAGES**

List<`ccrz.cc_bean_Message`>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

**ccrz.ccApi.SUCCESS**

Boolean


Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the created cart.


**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the created cart.

 **Tip:** If this key is empty or missing from the return data, an error likely occurred. [Consider rolling back your operation](#).

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the created `ccrz__E_Cart__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

### Example: Example

Create a cart from values specified in `ccrz.ccApiCart.CART_OBJLIST`, and include the created cart in the return data.

```
Boolean wasSuccessful = false;
Map<String, Object> cartToCreate = new Map<String, Object> {
    'cartType' => 'Cart'
    'activeCart' => true,
    'name' => 'descriptiveCartName',
    'storefront' => 'storefrontName',
};

Map<String, Object> cartCreateInputData = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiCart.CART_OBJLIST => new List<Map<String, Object>> {cartToCreate},
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_REFETCH => true
        }
    }
};

try {
    Map<String, Object> createCartResults = ccrz.ccApiCart.create(cartCreateInputData);

    wasSuccessful = (Boolean)createCartResults.get(ccrz.ccApi.SUCCESS);
    List<Object> createdCartList = (List<Object>)
createCartResults.get(ccrz.ccApiCart.CART_OBJLIST);
    Map<String, Object> createdCart = (Map<String, Object>) createdCartList[0];
    // Do something with the created cart
} catch (Exception e) {
    // Error handling...
}
```


## ccrz.ccApiCart.createCartItemGroup

Creates a shipping group of cart items for line-level independence (LLI) checkout.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> createCartItemGroup(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

[ccrz.ccLogicCartCreateCartItemGroup](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:


### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### '`ECartItemGroups`'

`List<Map<String, Object>>`, where each `Map<String, Object>` represents the [ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#) record to create. This map can specify subkeys for field values to set on the cart item group record. For a list of fields, see the [ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#) reference.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes [ccrz.ccApiCart.fetch](#) for returning the updated cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the updated `ccrz__E_Cart__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

**ccrz.ccApiCart.ITEMGROUPID**

String that specifies the Salesforce ID of the created cart item group.

 **Example: Example**

Create a cart item group from input values.

```
Boolean wasSuccessful = false;
Date requestedDate = date.today();

Map<String, Object> cartItemGroupToCreate = new Map<String, Object> {
    'cart' => 'cartID'
    'note' => 'Text for the shipping note',
    'requestedDate' => requestedDate,
    'shipTo' => 'contactAddressID',
    'shipAmount' => '20.00',
    'shipMethod' => 'Ground'
};

Map<String, Object> cartItemGroupCreateInputData = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    'ECartItemGroupss' => new List<Map<String, Object>> {cartItemGroupToCreate}
};

try {
    Map<String, Object> createCartItemGroupResults =
    ccrz.ccApiCart.createCartItemGroup(cartItemGroupCreateInputData);
    wasSuccessful = (Boolean)createCartItemGroupResults.get(ccrz.ccApi.SUCCESS);
    String newItemGroupId = (String)
    createCartItemGroupResults.get(ccrz.ccApiCart.ITEMGROUPID);
    // Do something with the created cart item group
} catch (Exception e) {
```



```
// Error handling...
}
```

SEE ALSO:

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)


## ccrz.ccApiCart.createWishList

Creates a wish list and wish list items based on a specific cart ID and its cart items.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> createWishList (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider


[ccrz.ccLogicCartCreateWL](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

**ccrz.ccApiCart.CART\_ID**

The Salesforce ID of the cart to create a wish list from.

**ccrz.ccApiCart.CART\_ENCID**

The encrypted ID of the cart to create a wish list from.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiWishList.fetch` for returning the created wish list.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiWishList.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiWishList.ID**

String that specifies the Salesforce ID of the created wish list.

SEE ALSO:

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)


## ccrz.ccApiCart.fetch

Executes a SOQL query of cart records, and returns a list of carts that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider

`ccrz.ccServiceCart`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

#### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.


**ccrz.ccApiCart.ACTIVECART**

Boolean

Value	Usage
<i>true</i>	Return the active cart for a specific owner. Use this key with when you also specify <code>ccrz.ccApiCart.BYOWNER</code> .
<i>false</i> (default)	Return all carts that match the other query criteria.

**ccrz.ccApiCart.BYOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

 **Note:** Sharing rules in your org can affect what data you can see when you filter by cart owner.

**ccrz.ccApiCart.BYUSER**

String that specifies the ID of a specific user. Return only cart records where the User field matches this user ID.

 **Note:** Sharing rules in your org can affect what data you can see when you filter by user.

**ccrz.ccApiCart.BYSTOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiCart.CART\_ENCID**


String that specifies the encrypted ID of the cart to query.

**ccrz.ccApiCart.CART\_ENCIDLIST**

Set&lt;String&gt; of encrypted cart IDs to query.

**ccrz.ccApiCart.CART\_IDLIST**


Set&lt;String&gt; of cart IDs to query.

 **Note:** This key takes precedence over `ccrz.ccApiCart.CART_ENCID` or `ccrz.ccApiCart.CART_ENCIDLIST`. If you specify both `ccrz.ccApiCart.CART_IDLIST` and `ccrz.ccApiCart.CART_ENCIDLIST`, this method ignores `ccrz.ccApiCart.CART_ENCIDLIST`.

**ccrz.ccApiCart.CARTNAME**String that specifies a cart name to query. This value corresponds to the `Cart Auto Number (Name)` field on the cart record.**ccrz.ccApiCart.CARTSTATUS**String that specifies a cart status for filtering records in the return data. This value corresponds to the `ccrz__CartStatus__c` field on the cart record.**ccrz.ccApiCart.CARTTYPE**

Deprecated. The managed package no longer references this key.

**ccrz.ccService.SEARCHSTRING**String that specifies filter criteria to pass to the `ccrz.ccApiCart.fetchCartItems` method.

 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

## Outputs


Map<String, Object> that can include the following keys:

### ccrz.ccApi.API\_VERSION

Integer that indicates which API version was used for the query.


### ccrz.ccApi.SUCCESS

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is false, consider rolling back the API transaction to a previous savepoint.

### ccrz.ccApiAddress.ADDRESSLIST

List<Map<String, Object>>, where each Map<String, Object> represents a [ccrz\\_\\_E\\_ContactAddr\\_\\_c](#) associated with the cart's shipping or billing addresses. For more information about fetching address data, see [ccrz.ccApiAddress.fetch](#).

 **Tip:** To specify which fields the API returns for each contact address, [request a specific data size](#) for the `ccrz.ccApiAddress.ENTITYNAME` key in your input data. You can request unique sizes for cart data and contact address data:


```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    },
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

You can also exclude the address records from the return data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
        ccrz.ccApi.SZ_ASSC => false
    }
}
```

### ccrz.ccApiCart.CART\_OBJLIST

List<Map<String, Object>>, where each Map<String, Object> represents a [ccrz\\_\\_E\\_Cart\\_\\_c](#) record that matches the query. By default, this list sorts by each cart's last modified date in descending order.

 **Tip:** To specify which fields the API returns for each cart record, [request a specific data size](#) for the `ccrz.ccApiCart.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
```

```


        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}

```

For information about which fields are available with different data sizes, see the [ccrz\\_\\_E\\_Cart\\_\\_c](#) reference.

### **ccrz.ccApiProduct.PRODUCTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [ccrz\\_\\_E\\_Product\\_\\_c](#) associated with the cart's items. For more information about fetching product data, see [ccrz.ccApiProduct.fetch](#).

 **Tip:** To specify which fields the API returns for each product record, [request a specific data size](#) for the `ccrz.ccApiProduct.ENTITYNAME` key in your input data. You can request unique sizes for cart data and product data:

```

ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    },
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}

```

You can also exclude the product records from the return data:

```

ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
        ccrz.ccApi.SZ_ASSC => false
    }
}

```

### **Example: Examples**

Query a pair of cart IDs, and capture the first cart's status in a variable.

```

Set<String> cartIdList = new Set<String>{'CART_ID_1', 'CART_ID_2'};

Map<String, Object> cartFetchQuery = new Map<String, Object>{
    ccrz.ccApiCart.CART_IDLIST => cartIdList,
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
};

try {
    Map<String, Object> cartFetchData = ccrz.ccApiCart.fetch(cartFetchQuery);
    if (cartFetchData.get(ccrz.ccApiCart.CART_OBJLIST) != null) {
        List<Map<String, Object>> outputCartList = (List<Map<String, Object>>)
cartFetchData.get(ccrz.ccApiCart.CART_OBJLIST);
        String firstCartStatus = (String) outputCartList[0].get('cartStatus');
    }
} catch (Exception e) {
    // Error handling...
}

```

Query all carts with a status of Abandoned.

```
Map<String, Object> cartFetchQuery = new Map<String, Object>{
    ccrz.ccApiCart.CARTSTATUS => 'Abandoned',
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
};

try {
    Map<String, Object> cartFetchData = ccrz.ccApiCart.fetch(cartFetchQuery);
    if (cartFetchData.get(ccrz.ccApiCart.CART_OBJLIST) != null) {
        List<Map<String, Object>> outputCartList = (List<Map<String, Object>>)
cartFetchData.get(ccrz.ccApiCart.CART_OBJLIST);
    }
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

## ccrz.ccApiCart.fetchCartItems

Executes a SOQL query of cart item records, and returns a list of cart items that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetchCartItems (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceCartItem`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### `ccrz.ccApiCart.CART_ENCIDLIST`

`Set<String>` of encrypted cart IDs to query.

#### `ccrz.ccApiCart.CART_IDLIST`

`Set<String>` of cart IDs to query.



**Note:** Include `ccrz.ccApiCart.CART_ENCIDLIST` or `ccrz.ccApiCart.CART_IDLIST`, but not both.

### Inputs (Optional)

The input map can also include the following keys:

#### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

#### `ccrz.ccApiCart.CARTITEM_IDS`

`List<String>` of Salesforce IDs for specific `ccrz__E_CartItem__c` records to query.



**Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

#### `ccrz.ccApiCart.CARTITEM_INCLUDE_CHILDREN`

Boolean



**Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

Value	Usage
<code>true</code> (default)	Return child <code>ccrz__E_CartItem__c</code> records (minor line items) associated with parent <code>ccrz__E_CartItem__c</code> records (major line items) requested in <code>ccrz.ccApiCart.CARTITEM_IDS</code> .
<code>false</code>	Return only the <code>ccrz__E_CartItem__c</code> records requested in <code>ccrz.ccApiCart.CARTITEM_IDS</code> . Exclude any child <code>ccrz__E_CartItem__c</code> records that exist from the return data.



**ccrz.ccApiCart.CISERVSPLIT**

Integer that specifies a buffer size of cart items to process at one time. The default value is 200.

**ccrz.ccService.SEARCHSTRING**

String for filtering cart items using the SOQL `LIKE` operator. You can specify wildcard characters that the `LIKE` operator supports. By default, the `ccrz.ccServiceCartItem` data service provider class specifies that the search query applies to the following `ccrz__E_CartItem__c` object fields:

- `ccrz__Comments__c`
- `ccrz__ExtName__c`
- `ccrz__ExtSKU__c`
- `ccrz__SortName__c`
- `ccrz__Product__r.Name`
- `ccrz__Product__r.SKU__c`

Assume a logical `OR` operator between fields.



**Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

## Outputs


Map<`String`, `Object`> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiCart.CARTITEMATTRIDS**

Set<`String`> of attribute IDs that correspond to the returned cart items.


**ccrz.ccApiCart.CARTITEMPRODUCTIDS**

Set<`String`> of product IDs that correspond to the returned cart items.

**ccrz.ccApiCart.CARTITEMSBYID**

Data from the `ccrz__E_CartItem__c` records that match the fetch query. This return data can take the following types, depending on values specified in the input map:

Type	Input Map Criteria
Map<String, List<Map<String, Object>>>, keyed by parent ccrz__E_Cart__c, where each Map<String, Object> represents a ccrz__E_CartItem__c	Default return format
Map<String, List<Map<String, Object>>>, keyed by parent ccrz__E_Cart__c, where each Map<String, Object> is determined by ccrz.ccServiceCartItem.getGroupByMap (Map<String, Object>). This data service provider method can request querying against an aggregate function.	<pre>ccrz.ccApi.SIZING =&gt; new Map&lt;String, Object&gt; {     ccrz.ccApiCart.CLI_ENTITYNAME =&gt; new Map&lt;String, Object&gt; {         ccrz.ccApi.SZ_DATA =&gt;         ccrz.ccApi.SZ_GROUPBY     } }</pre>
Map<String, List<ccrz__E_CartItem__c>>	<pre>ccrz.ccApi.SIZING =&gt; new Map&lt;String, Object&gt; {     ccrz.ccApiCart.CLI_ENTITYNAME =&gt; new Map&lt;String, Object&gt; {         ccrz.ccApi.SZ_SKIPTRZ =&gt; true     } }</pre>

 **Tip:** To specify which fields the API returns for each cart record, [request a specific data size](#) for the ccrz.ccApiCart.CLI\_ENTITYNAME key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.CLI_ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)  
[Comparison Operators](#)  
[Logical Operators](#)


## ccrz.ccApiCart.getActive

Calls the ccrz.ccApiCart.fetch method and returns the active cart for the current account from the set of queried carts. This method sets other carts that the account can access to inactive.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> getActive(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicCartGetActive`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### `ccrz.ccApiCart.BYEFFECTIVEACCOUNT`

Boolean

Value	Usage
<code>true</code>	Filter the cart query by the selected effective account. This method evaluates this key only when your storefront uses the <a href="#">Cart per Account mode for effective accounts</a> .
<code>false</code> (default)	Ignore effective accounts.

**ccrz.ccApiCart.BYOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

**Note:**

- Sharing rules in your org can affect what data you can see when you filter by cart owner.
- If you don't include this key, this method falls back to `ccrz.cc_CallContext.currUserId`.

**ccrz.ccApiCart.BYSTOREFRONT**

String that specifies the name of a storefront for filtering the query.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to query.

**ccrz.ccApiCart.CART\_IDLIST**

Set<String> of Salesforce IDs of carts to query.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to query.

**ccrz.ccApiCart.CART\_ENCIDLIST**

Set<String> of encrypted cart IDs to query.



**Note:** If you don't specify `ccrz.ccApiCart.CART_ID`, `ccrz.ccApiCart.CART_IDLIST`, `ccrz.ccApiCart.CART_ENCID`, or `ccrz.ccApiCart.CART_ENCIDLIST`, this method falls back to `ccrz.cc_CallContext.currCartId`.

**ccrz.ccApiCart.CURRCODE**

String that specifies an ISO 4217 currency code, such as *USD* or *JPY*.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userCurrency`.

## Outputs

Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.MESSAGES**

List<`ccrz.cc_bean_Message`>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where each Map<String, Object> represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 **Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[Salesforce Help Manage Carts for Effective Accounts](#)


## ccrz.ccApiCart.getDeliveryDates

Calls the `ccrz.cc_api_DeliveryDate` extension point class and returns the available delivery dates for one or more shipping addresses.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> getDeliveryDates (Map<String, Object>)
```

### Service Layer Classes

**Logic Service Provider**

[ccrz.ccLogicCartGetDeliveryDates](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:


### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.


### `ccrz.cc_api_DeliveryDate.CART_ID`

String that specifies the ID of the cart record used for checkout.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCart.SHIPPING_ADDRLIST`

`List<Map<String, Object>>` where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c` record with shipping address details.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

In each `Map<String, Object>`, you can specify values for the following `ccrz__E_ContactAddr__c` fields:

Field	Data Type
<code>addressFirstline</code>	String
<code>addressSecondline</code>	String
<code>addressThirdline</code>	String
<code>addrReadOnly</code>	Boolean
<code>city</code>	String
<code>companyName</code>	String
<code>country</code>	String
<code>countryISOCCode</code>	String
<code>daytimePhone</code>	String
<code>email</code>	String
<code>firstName</code>	String
<code>homePhone</code>	String
<code>lastName</code>	String
<code>middleName</code>	String
<code>postalCode</code>	String
<code>sfid</code>	String
<code>state</code>	String

Field	Data Type
stateISOCode	String

## Outputs


Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

### **ccrz.ccApi.SUCCESS**


Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### **ccrz.ccApiCart.SHIPPING\_ADDRLIST**

Map<String, Object> where:

- The key is the contact address `sfid`.
- The object is a Map<String, Object> that the `ccrz.cc_api_DeliveryDate.getDeliveryDates` method returns.

 **Note:** This method also directly returns the Map<String, Object> output from the `ccrz.cc_api_DeliveryDate.getDeliveryDates` method.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz.cc\\_api\\_DeliveryDate.getDeliveryDates](#)

[Salesforce Help Specify Shipping Options for Standard Checkout](#)


## ccrz.ccApiCart.getShippingOptions

Calls the `ccrz.cc_api_ShippingAndHandling` extension point class and returns the available shipping options for one or more shipping addresses.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> getShippingOptions(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

[ccrz.ccLogicCartGetShippingOptions](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCart.SHIPPING_ADDR`

`Map<String, Object>` that represents a `ccrz__E_ContactAddr__c` record with shipping address details.

### `ccrz.ccApiCart.SHIPPING_ADDRLIST`

`List<Map<String, Object>>` where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c` record with shipping address details.

Whether you specify `ccrz.ccApiCart.SHIPPING_ADDR` or `ccrz.ccApiCart.SHIPPING_ADDRLIST`, include the following `ccrz__E_ContactAddr__c` fields and values in each `Map<String, Object>`:

Field	Data Type
countryISOCODE	String
postalCode	String



Field	Data Type
sfid	String
stateISOCode	String

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApiCart.CART\_ID**

String that specifies the ID of the cart used for checkout.

## Outputs


Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### **ccrz.ccApiCart.SHIPPING\_OPTIONS**

The shipping options for the specified contact address or addresses. This return data can take the following types, depending on values specified in the input map:

Type	Input Map Criteria
Map<String, Object>	ccrz.ccApiCart.SHIPPING_ADDR
List<Map<String, Object>>	ccrz.ccApiCart.SHIPPING_ADDRLIST

For each Map<String, Object> returned:

- The key is the contact address `sfid`.

- The object is a serialized instance of `ccrz.cc_ctrl_hlpr_ShippingOption`, which the `ccrz.cc_api_ShippingAndHandling.getShippingOptions` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz.cc\\_api\\_ShippingAndHandling.getShippingOptions](#)

[Salesforce Help Specify Shipping Options for Standard Checkout](#)


## ccrz.ccApiCart.initItemGroups

Initializes shipping groups on a specific cart for line-level independence (LLI) checkout. This method creates a single default shipping group with all cart items and the account's default shipping address. If the cart already contains at least one group, this method assigns any cart items that aren't assigned to a group to that group.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> initItemGroups (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

[ccrz.ccLogicCartInitItemGroups](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

**ccrz.ccApiCart.CART\_ID**

The Salesforce ID of the cart to initialize shipping groups for.

**ccrz.ccApiCart.CART\_ENCID**

The encrypted ID of the cart to initialize shipping groups for.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the updated cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean


Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiCart.CART\_ENCID**

The encrypted ID of the updated cart.

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the updated `ccrz__E_Cart__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

 **Example: Example**

Initialize shipping groups for a specific cart.

```
Map<String, Object> cartItemGroupInputData = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiCart.CART_ENCID => 'encrypted cart ID',
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_REFETCH => true
        }
    }
};

try {
    Map<String, Object> initItemGroupsResults =
ccrz.ccApiCart.initItemGroups(cartItemGroupInputData);
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

**ccrz.ccApiCart.place**

Converts a cart to a corresponding order synchronously.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> place(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

```
ccrz.ccLogicCartPlace
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the cart to convert.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to convert.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the ordered cart.


```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

### `ccrz.ccApiCart.ENFORCE_PRICE_VALIDATION_ON_PLACE`

Boolean




**Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

Value	Usage
<i>true</i>	Evaluate whether to reprice the cart before starting the order placement process. If the cart's price data isn't current, this method doesn't create an order from the cart and returns a <code>ccrz.cc_api_Cart.CartValidationException</code> .   <b>Note:</b> When you enable Cart Scale, this input is <i>true</i> by default.
<i>false</i> (default)	Don't evaluate the cart's price data before starting order placement.

**ccrz.ccApiCart.FORCE\_PLACE\_SYNC**

Boolean

 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

Value	Usage
<i>true</i>	Place the order synchronously, regardless of the value of the <code>ccrz.ccApiCart.PLACE_ASYNC</code> input.
<i>false</i> (default)	Default logic determines whether to place the order synchronously or asynchronously.

**ccrz.ccApiCart.PAYMENTDATA**


JSON-formatted string that describes payment information passed to the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA` input.

**ccrz.ccApiCart.PAYMENTRESULT**

`Map<String, Object>` that represents the output from the `ccrz.cc_hk_Payment.processPayment` extension point method.

**ccrz.ccApiCart.PLACE\_ASYNC**

Boolean

 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).


Value	Usage
<i>true</i>	Place the order asynchronously by calling the <code>ccrz.ccApiCart.placeAsync</code> method.
<i>false</i> (default)	Place the order synchronously.

 **Note:**

- By default, this flag is ignored for a guest buyer checkout or when the `ccrz.ccApiCart.FORCE_PLACE_SYNC` input is *true*.
- For storefronts that haven't enabled cart scale, order placement is always synchronous.
- For more information about whether order placement executes synchronously or asynchronously, see [ccrz.ccLogicCartPlace.checkCartForAsyncPlace](#).

**ccrz.ccApiCart.SKIP\_VALID\_CHECK**

Boolean

Value	Usage
<i>true</i>	<p>During this method's execution, ignore the cart's <code>ccrz__ValidationStatus__c</code> field and attempt to create an order from the cart.</p> <p> <b>Important:</b> When you pass <code>ccrz.ccApiCart.SKIP_VALID_CHECK =&gt; true</code>, call the <code>ccrz.ccApiCart.price</code> and <code>ccrz.ccApiCart.validate</code> methods to ensure that the cart items and prices are valid.</p>
<i>false</i> (default)	<p>During this method's execution, verify that the cart's <code>ccrz__ValidationStatus__c</code> field is set before creating an order for this cart. When this field is set, all items are valid and have a price, and the buyer is allowed to check out from the Shopping Cart page.</p>

**ccrz.ccApiOrder.PARAM\_ORDER\_STATUS**

String that specifies the status of the corresponding order when it's created. By default, this method sets the value *Order Submitted*.

**ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

 **Note:**

- Create the payment information on the cart *before* you call this method, and then associate the payment information with the order *after* this method completes.
- If you specify both `ccrz.ccApiCart.PAYMENTDATA` and `ccrz.ccApiCart.TRANSPAYMENTDATA` and these keys describe a stored payment, this method creates a corresponding `ccrz__E_StoredPayment__c` record.
- If you specify `ccrz.ccApiCart.PAYMENTDATA`, `ccrz.ccApiCart.TRANSPAYMENTDATA`, and `ccrz.ccApiCart.PAYMENTRESULT`, this method creates a corresponding `ccrz__E_TransactionPayment__c` record.

## Outputs

Map<*String*, *Object*> that can include the following keys:


**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, which contains one Map<String, Object> that represents the `ccrz__E_Cart__c` record, which the `ccrz.ccApiCart.fetch` method returned.

**Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

**ccrz.ccApiCart.ORDER\_ID**

String that specifies the Salesforce ID of the created `ccrz__E_Order__c`.



**Important:** In B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12):

- The `ccrz.ccLogicCartPlace` logic service provider class includes methods that replace the `ccrz.cc_api_Cart.placeEOrder` public method, which is deprecated in this version.
- As a result, the `ccrz.ccLogicCartPlace` logic service provider executes differently compared to B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11) or older. The `ccrz.cc_api_Cart.placeEOrder` public method remains available for those older versions.
- To continue using `ccrz.cc_api_Cart.placeEOrder` logic for placing an order, set the Checkout storefront configuration module's Use Order Hook setting (`co.usehkorde`) to `TRUE`. This configuration specifies that methods of the [ccrz.cc\\_hk\\_Order](#) extension point class execute and call `ccrz.cc_api_Cart.placeEOrder`. When this configuration is `FALSE`, the `ccrz.ccLogicCartPlace` logic service provider executes instead.

## SEE ALSO:

[ccrz.ccApiCart.price](#)

[ccrz.ccApiCart.validate](#)

[ccrz.cc\\_hk\\_Payment.processPayment](#)

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

**ccrz.ccApiCart.placeAsync**

Converts a cart to a corresponding order using asynchronous Apex jobs. This method is called recursively for processing multiple, small batches of cart items. When this method is first called, it first executes synchronous logic, and then enqueues recursive calls that execute asynchronous logic. Only the synchronous logic provides return data.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global static Map<String, Object> placeAsync (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

[ccrz.ccLogicCartPlaceAsync](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the cart to convert.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to convert.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiCart.ALLOW_PRICING_ON_ASYNC_ORDER`


Boolean

Value	Usage
<code>true</code>	<p>If the <code>ccrz.ccApiCart.ENFORCE_PRICE_VALIDATION_ON_PLACE</code> input is <code>true</code> and the cart's price data isn't current, call the <a href="#">ccrz.ccApiCart.priceAsync</a> method to reprice the cart.</p> <p> <b>Note:</b> This behavior can cause issues with the cart's shipping and taxes, which are calculated based on the cart's previous price data, not the repriced cart. The</p>

Value	Usage
	<code>ccrz.ccApiCart.priceAsync</code> method doesn't update shipping or taxes. Consider creating an extension that updates shipping and tax calculations for asynchronous pricing.
<code>false</code> (default)	If the cart's price data isn't current, the cart isn't repriced, and the order placement process fails.


**ccrz.ccApiCart.CARTITEMLASTID**

String that specifies the Salesforce ID of the final major cart item that was processed during the previous recursive job. This ID is used only during recursive jobs that process cart items.

 **Note:** To determine the set of cart items in the next asynchronous job and return a consistently ordered list, the `ccrz.ccLogicCartPlaceAsync.retrieveCartItems` method queries major cart items with a `SQL ORDER BY Id` clause.

**ccrz.ccApiCart.CART\_PLACE\_BATCH\_SIZE**


Integer that specifies the number of major cart items to process during each recursive job.

 **Note:**

- If you don't specify a value for this input, the method defaults to 200 major cart items.
- The job also queries any minor cart items that are associated with the major cart items. As a result, the total number of processed `ccrz__E_CartItem__c` records is higher than the specified batch size when the cart contains complex products.

**ccrz.ccApiCart.ENFORCE\_PRICE\_VALIDATION\_ON\_PLACE**

Boolean

Value	Usage
<code>true</code>	Evaluate whether to reprice the cart before starting the order placement process. If the cart's price data isn't current, this method doesn't create an order from the cart and returns a <code>ccrz.cc_api_Cart.CartValidationException</code> .   <b>Note:</b> When you enable Cart Scale, this input is <code>true</code> by default.
<code>false</code> (default)	Don't evaluate the cart's price data before starting order placement.

**ccrz.ccApiOrder.PARAM\_ORDER\_STATUS**

String that specifies the status of the corresponding order when it's created. By default, this method sets the value `Order Submitted`.

**ccrz.ccApiCart.PAYMENTDATA**

JSON-formatted string that describes payment information passed to the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA` input.

**ccrz.ccApiCart.PAYMENTRESULT**

`Map<String, Object>` that represents the output from the `ccrz.cc_hk_Payment.processPayment` extension point method.


**ccrz.ccApiCart.PLACE\_ASYNC\_FINISH**

Boolean

Value	Usage
<i>true</i>	The current method execution is a recursive, asynchronous call that continues the order placement process after the initial call.
<i>false</i> (default)	The current method execution is the synchronous portion of the initial call that begins the order placement process.


**ccrz.ccApiCart.PRICING\_FOR\_ORDER\_FAILED**

Boolean

Value	Usage
<i>true</i>	An error occurred when pricing the cart during the previous iteration of the recursive, asynchronous call. The <code>ccrz.ccLogicCartPlaceAsync.updateOrderOnError</code> method executes.   <b>Note:</b> This input is used only when the <code>ccrz.ccApiCart.ALLOW_PRICING_ON_ASYNC_ORDER</code> input is <i>true</i> and the cart was repriced because its price data wasn't current.
<i>false</i> (default)	No error occurred.

**ccrz.ccApiCart.SKIP\_VALID\_CHECK**

Boolean

Value	Usage
<i>true</i>	During this method's execution, ignore the cart's <code>ccrz__ValidationStatus__c</code> field and attempt to create an order from the cart.   <b>Important:</b> When you pass <code>ccrz.ccApiCart.SKIP_VALID_CHECK =&gt; true</code> , call the <code>ccrz.ccApiCart.price</code> and <code>ccrz.ccApiCart.validate</code> methods to ensure that the cart items and prices are valid.
<i>false</i> (default)	During this method's execution, verify that the cart's <code>ccrz__ValidationStatus__c</code> field is set before creating an order for this cart. When this field is set, all items are valid and have a price, and the buyer is allowed to check out from the Shopping Cart page.

**ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

 **Note:**

- Create the payment information on the cart *before* you call this method, and then associate the payment information with the order *after* this method completes.
- If you specify both `ccrz.ccApiCart.PAYMENTDATA` and `ccrz.ccApiCart.TRANSPAYMENTDATA` and these keys describe a stored payment, this method creates a corresponding `ccrz__E_StoredPayment__c` record.

- If you specify `ccrz.ccApiCart.PAYMENTDATA`, `ccrz.ccApiCart.TRANSPAYMENTDATA`, and `ccrz.ccApiCart.PAYMENTRESULT`, this method creates a corresponding `ccrz__E_TransactionPayment__c` record.

## Outputs

Map<**String**, **Object**> that can include the following keys:




**Note:** Only the initial synchronous `ccrz.ccApiCart.placeAsync` method call includes return data. The return data for subsequent asynchronous Apex jobs isn't accessible.

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiCart.ORDER_ID`

String that specifies the Salesforce ID of the created `ccrz__E_Order__c`.

SEE ALSO:

[ccrz.ccApiCart.priceAsync](#)

[ccrz.ccApiCart.validate](#)

[ccrz.cc\\_hk\\_Payment.processPayment](#)

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

## `ccrz.ccApiCart.price`


Reprices all the items in a cart.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> price(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccLogicCartPrice`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted cart ID to price.

### `ccrz.ccApiCart.CART_ID`

String that specifies the cart ID to price.

 **Note:** Include `ccrz.ccApiCart.CART_ENCID` or `ccrz.ccApiCart.CART_ID`, but not both.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the repriced cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
```

```

        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}

```

**ccrz.ccApiCart.CILIST**

Map<String, Map<String, Object>> of the cart items to reprice. For example, when a buyer updates quantities of items in the cart, this list specifies the cart items that have changed. This map is keyed by cart item ID, and each nested Map<String, Object> defines the following subkeys:

**ccrz.ccApiCart.CIQTY**

Integer that specifies the item's quantity.

**ccrz.ccApiCart.CICMTS**

String that specifies the item's comments.

**ccrz.ccApiCart.CIPREPROC**

Boolean

Value	Usage
<i>true</i>	Call the <code>ccrz.cc_api_CartExtension.preprocess</code> extension point method for processing the cart items.
<i>false</i> (default)	Skip the <code>ccrz.cc_api_CartExtension.preprocess</code> extension point method.

**ccrz.ccApiCart.FORCE\_PRICE\_SYNC**

Boolean

 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

Value	Usage
<i>true</i>	Execute pricing synchronously, regardless of the value of the <code>ccrz.ccApiCart.PRICE_ASYNC</code> input.
<i>false</i> (default)	Default logic determines whether to price the cart synchronously or asynchronously.

**ccrz.ccApiCart.PARTIAL\_CART\_UPDATE**

Boolean


 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

Value	Usage
<i>true</i>	Apply pricing, entitlements, and quantity updates only to a certain subset of cart items. The <code>ccrz.ccLogicCartPrice.retrieveCartData</code> method defines the business logic for the partial update.
<i>false</i> (default)	Execute default pricing logic.

**ccrz.ccApiCart.PRICE\_ASYNC**

Boolean

 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

Value	Usage
<code>true</code>	Execute pricing asynchronously by calling the <code>ccrz.ccApiCart.priceAsync</code> method.   <b>Note:</b> By default, this flag is ignored for a guest buyer's cart, for partial cart updates, or when the <code>ccrz.ccApiCart.FORCE_PRICE_SYNC</code> input is <code>true</code> .
<code>false</code> (default)	Execute pricing synchronously.

**ccrz.ccApiCart.SKIP\_CART\_PRICING**

Boolean

Value	Usage
<code>true</code>	Skip the pricing logic defined in the <code>ccrz.ccLogicCartPrice</code> logic service provider.
<code>false</code> (default)	Execute the pricing logic defined in <code>ccrz.ccLogicCartPrice</code> .

**ccrz.ccApiCart.SYNC\_MAX\_CART\_SIZE**

Integer that specifies the maximum number of items that must be in a cart for synchronous processing. This value overrides the value of the Asynchronous Processing Threshold storefront configuration setting in the Cart Scale module (`cs.asyncthreshold`).

 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

## Outputs


Map<`String`, `Object`> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart that was priced.



**Tip:** If this key is empty or missing from the return data, an error likely occurred. [Consider rolling back your operation.](#)

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where each Map<String, Object> represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.



**Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.cc\\_api\\_CartExtension](#)

## ccrz.ccApiCart.priceAsync

Reprices all the items in a cart using asynchronous Apex jobs. This method supports pricing a cart with a much larger number of items than the `ccrz.ccApiCart.price` method. This method also includes both a synchronous portion that updates the cart's status and instantiates the asynchronous reprice jobs. However, the asynchronous behavior means that there's no guarantee for how long it takes for the jobs to complete.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global static Map<String, Object> priceAsync (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

[ccrz.ccLogicCartPriceAsync](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:



**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted cart ID to price.

**ccrz.ccApiCart.CART\_ID**

String that specifies the cart ID to price.

 **Note:** Include `ccrz.ccApiCart.CART_ENCID` or `ccrz.ccApiCart.CART_ID`, but not both.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApiCart.CART\_PRICE\_BATCH\_SIZE**


Integer that specifies the number of major cart items to price during each asynchronous Apex job.

 **Note:**

- If you don't specify a value for this input, the method defaults to 200 major cart items.
- The job prices any minor cart items that are associated with the major cart items. As a result, the total number of priced `ccrz__E_CartItem__c` records is higher when the cart contains complex products. When a pricing scenario includes a large number of complex products, consider specifying a lower batch size to help fine-tune performance.

**ccrz.ccApiCart.CARTITEMLASTID**

String that specifies the Salesforce ID of the final major cart item that was priced during the previous asynchronous Apex job.

 **Note:** To determine the set of cart items in the next asynchronous job and return a consistently ordered list, the `ccrz.ccLogicCartPriceAsync.retrieveCartItems` method queries major cart items with a `SOQL ORDER BY Id` clause.

**ccrz.ccApiCart.CONTINUE\_ASYNC\_PLACE\_AFTER\_PRICING**

Boolean

Value	Usage
<code>true</code>	This method is executing as part of the <code>ccrz.ccApiCart.placeAsync</code> method. When this method completes, it calls <code>ccrz.ccApiCart.placeAsync</code> recursively to continue the order placement process.
<code>false</code> (default)	This method is executing outside of the <code>ccrz.ccApiCart.placeAsync</code> method.

**ccrz.ccApiCart.PRICE\_ASYNC\_FINISH**

Boolean

Value	Usage
<code>true</code>	The current method execution is a recursive, asynchronous call that continues the cart pricing after the initial call.
<code>false</code> (default)	The current method execution is the synchronous portion of the initial call that begins the cart pricing process.

## Outputs

Map<[String](#), [Object](#)> that can include the following keys:



**Note:** Only the synchronous portion of the `ccrz.ccApiCart.priceAsync` method call includes return data. The return data for the asynchronous Apex jobs isn't accessible.

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart that was priced.



**Tip:** If this key is empty or missing from the return data, an error likely occurred. [Consider rolling back your operation](#).

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## `ccrz.ccApiCart.processItemGroups`


Moves cart items between shipping groups for line-level independence (LLI) checkout.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> processItemGroups (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

```
ccrz.ccLogicCartProcessItemGroups
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:


### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiCart.CART_ENCID`

The encrypted ID of the cart.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCart.MOVEDATA`

`List<Map<String, Object>>` where each `Map<String, Object>` contains the following keys that describe the cart items to move.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCart.ITEMID`

String that specifies the Salesforce ID of the `ccrz__E_CartItem__c` to move.

### `ccrz.ccApiCart.TARGETQTY`

Decimal that specifies the quantity of the `ccrz__E_CartItem__c` to move.

 **Note:** If you don't specify a quantity, this method moves the full quantity.

**ccrz.ccApiCart.TARGETGROUP**

String that specifies the Salesforce ID of the `ccrz__E_CartItemGroup__c` to move `ccrz__E_CartItem__c` records to.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.



**Note:** If `ccrz.ccApiCart.TARGETGROUP` specifies the same cart item group that the `ccrz.ccApiCart.MOVEDATA` cart items currently belong to, this method doesn't move the cart items.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApiCart.MERGELINES**

Boolean

Value	Usage
<code>true</code>	Combine quantities of the same SKU are combined into a single line item in the target cart item group. For usage, see <a href="#">Specify Shipping Group Options for LLI Checkout</a> .
<code>false</code> (default)	Leave quantities of the same SKU on separate line items in the target cart item group.



**Note:** For a dynamic kit, subscription, or externally priced product, this method can move quantities to another group, but can't combine quantities.

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**Example: Example**

Move two cart items to a specified shipping group, and don't combine quantities of the same SKU in the target group.

```
Map<String, Object> cartItemsToMove = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
```

```

ccrz.ccApiCart.CART_ENCID => 'encrypted cart ID',
ccrz.ccApiCart.TARGETGROUP => 'cart item group ID',
ccrz.ccApiCart.MERGELINES => false,
ccrz.ccApiCart.MOVEDATA => new List<Map<String, Object>> {
    new Map<String, Object> {
        ccrz.ccApiCart.ITEMID => 'cart item 1 ID',
        ccrz.ccApiCart.TARGETQTY => 1
    },
    new Map<String, Object> {
        ccrz.ccApiCart.ITEMID => 'cart item 2 ID',
        ccrz.ccApiCart.TARGETQTY => 1
    }
}
};

try {
    Map<String, Object> processItemGroupsResults =
ccrz.ccApiCart.processItemGroups(cartItemsToMove);
} catch (Exception e) {
    // Error handling...
}

```

SEE ALSO:

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)


## ccrz.ccApiCart.removeCart

Deletes cart records, including wish lists. If a deleted cart or wish list was active for the current account, this method also determines which cart or wish list becomes active next.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicCartRemoveCart`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include one of the following keys. If none of these keys are included, the method returns a `ccrz.ccApi.MissingInputException`.

#### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to delete.

#### `ccrz.ccApiCart.CART_ENCIDLIST`

`Set<String>` of encrypted IDs of the carts to delete.

#### `ccrz.ccApiCart.CART_ID`

String that specifies the ID of a cart to delete.

#### `ccrz.ccApiCart.CART_IDLIST`

`Set<String>` of IDs of carts to delete.

## Outputs

`Map<String, Object>` that can include the following keys:

#### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.


#### `ccrz.ccApi.MESSAGES`

`List<ccrz.cc_bean_Message>`, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

#### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

SEE ALSO:

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)


## ccrz.ccApiCart.removeCartItemGroup

Deletes shipping groups of cart items for line-level independence (LLI) checkout.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Service Layer Classes

#### Logic Service Provider

[ccrz.ccLogicCartRemoveItemGroups](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

**ccrz.ccApiCart.ITEMGROUPLIST**

List<[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)> of the cart item groups to delete.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Outputs

Map<[String](#), [Object](#)> that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.MESSAGES**

List<[ccrz.cc\\_bean\\_Message](#)>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

SEE ALSO:

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

## ccrz.ccApiCart.removeFrom


Removes items from a cart. This method doesn't invoke pricing logic for repricing the cart after the items have been removed.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> removeFrom (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicCartRemove`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to remove items from.

### `ccrz.ccApiCart.CART_ID`

String that specifies the ID of the cart to remove items from.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.


### `ccrz.ccApiCart.COUPON_CODE`

String that specifies a coupon code to remove from the cart.

 **Tip:** If you're not removing a coupon, specify some other string for this key, such as `true`.

### `ccrz.ccApiCart.LINE_DATA`

`List<ccrz.ccApiCart.LineData>` that describes the `ccrz__E_CartItem__c` records to remove from the cart. For each item, add a new `ccrz.ccApiCart.LineData` object to the list and set the `LineData.sfid` field cart item's Salesforce ID. For information about this list's keys and requirements, see [Inputs for Cart Line Data](#).

 **Tip:** This key accepts only the `List<ccrz.ccApiCart.LineData>` representation of line data, and doesn't accept the `List<Map<String, Object>>` representation that we recommend for the `ccrz.ccApiCart.addTo` method.

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint.</a>

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart that items were removed from.

 **Tip:** If this key is empty or missing from the return data, an error likely occurred. [Consider rolling back your operation.](#)

**ccrz.ccApiCart.repair**

Returns a cart where the status remains Repricing even after a repricing job completes, updates the status to Open, and resets the Repriced Date timestamp. A cart can remain in this status when a repricing job is unsuccessful or no longer available in the org. This method calls the `ccrz.ccApiCart.fetch` method for returning the affected cart.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

**Signature**

```
global static Map<String, Object> repair (Map<String, Object>)
```

**Service Layer Classes****Logic Service Provider**

`ccrz.ccLogicCartRepair`

**Inputs (Required)**

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include one of the following keys. If none of these keys are included, the method returns a `ccrz.ccApi.MissingInputException`.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to repair.

**ccrz.ccApiCart.CART\_ENCIDLIST**

Set<String> of encrypted IDs of the carts to repair.

**ccrz.ccApiCart.CART\_ID**

String that specifies the ID of a cart to repair.

**ccrz.ccApiCart.CART\_IDLIST**

Set<String> of IDs of carts to repair.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

 **Note:** By default, this method sets the following data size information:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_ASSC => false,
        ccrz.ccApi.SZ_REL => new List<String>
    }
}
```

## Outputs

Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.MESSAGES**

List<[ccrz.cc\\_bean\\_Message](#)>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is FALSE.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 **Example: Example**

Pass the encrypted ID of a cart to repair.

```
Map<String, Object> cartToRepair = new Map<String, Object>{
    ccrz.ccApiCart.CART_ENCID => 'CART_ENCID',
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
};

try {
    Map<String, Object> cartRepairResult = ccrz.ccApiCart.repair(cartToRepair);
    List<ccrz.cc_bean_Message> cartMessages =
(List<ccrz.cc_bean_Message>)ccrz.ccUtil.defv(cartRepairResult, ccrz.ccApi.MESSAGES,
new List<ccrz.cc_bean_Message>);
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

**ccrz.ccApiCart.revise**


Updates certain fields on a cart.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

 **Warning:** This method doesn't allow a guest user to update records. If you're creating subscriber code that requires a guest user to update records, you have the following options:

- If you're using B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11), invoke the [ccrz.ccLogic.updateObjects](#) utility method.
- Execute the DML operation in your custom Apex class using the [without sharing](#) accessor.

## Signature

```
global static Map<String, Object> revise (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

```
ccrz.ccServiceCart
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiCart.CART_OBJLIST`

`List<Map<String, Object>>`, which contains one `Map<String, Object>` that represents the `ccrz__E_Cart__c` record to update. You can update the values of only the following fields:

- `activeCart`
- `billTo`
- `buyerEmail`
- `buyerFirstName`
- `buyerLastName`
- `buyerPhone`
- `cartType`
- `name`
- `note`
- `sfid`

- shipAmount
- shipTo
- storefront

 **Note:** The `ccrz.ccServiceCart` data service provider determines which fields are editable.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the updated cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiCart.CART_OBJLIST`

`List<Map<String, Object>>` of the updated `ccrz__E_Cart__c` records.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

### Example: Example

Fetch a cart, and then update its name.

```
Boolean wasSuccessful = false;

Map<String, Object> cartFetchQuery = new Map<String, Object>{
```

```

ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
ccrz.ccApiCart.CART_IDLIST => new Set<String>{'CART_ID_1'},
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XL,
        ccrz.ccApi.SZ_REL => new List<String>(),
        ccrz.ccApi.SZ_ASSC => false
    }
}
};

try {
    Map<String, Object> cartFetchResult = ccrz.ccApiCart.fetch(cartFetchQuery);
    List<Object> fetchedCarts = (List<Object>)
cartFetchResult.get(ccrz.ccApiCart.CART_OBJLIST);
    Map<String, Object> fetchedCartMap = (Map<String, Object>) fetchedCarts[0];

    // Specify a new name
    fetchedCartMap.put('name', 'A perfectly cromulent cart');

    // Set the updated cart into a list for the revise method
    List<Map<String, Object>> cartToRevise = new List<Map<String, Object>>();
    cartToRevise.add(fetchedCartMap);

    // Call the revise method
    Map<String, Object> cartReviseResult = ccrz.ccApiCart.revise(new Map<String, Object>
{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiCart.CART_OBJLIST => cartToRevise
});

    wasSuccessful = (Boolean)cartReviseResult.get(ccrz.ccApi.SUCCESS);
} catch (Exception e) {
    // Error handling...
}

```

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccApiCart.reviseCartItemGroup


Updates certain fields on a cart item group record for line-level independence (LLI) checkout.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> reviseCartItemGroup (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider


ccrz.ccServiceCartItemGroup

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### '`ECartItemGroupsS`'

`List<Map<String, Object>>`, where each `Map<String, Object>` represents the [ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#) record to create. This map can specify subkeys for field values to set on the cart item group record. For a list of fields, see the [ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#) reference.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the updated cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```



## Outputs


Map<String, Object> that can include the following keys:

### ccrz.ccApi.API\_VERSION

Integer that indicates which API version was used for the query.


### ccrz.ccApi.SUCCESS

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is false, consider rolling back the API transaction to a previous savepoint.

### ccrz.ccApiCart.CART\_OBJLIST

List<Map<String, Object>>, where the Map<String, Object> represents the updated `ccrz__E_Cart__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

### Example: Example

Update values on an existing cart item group.

```
Boolean wasSuccessful = false;
Date requestedDate = date.today();

Map<String, Object> cartItemGroupToUpdate = new Map<String, Object> {
    'sfid' => 'Salesforce ID of a cart item group'
    'note' => 'Updated text for the shipping note',
    'requestedDate' => requestedDate,
    'shipTo' => 'contactAddressID',
    'shipAmount' => '40.00',
    'shipMethod' => 'Air'
};

Map<String, Object> cartItemGroupUpdateInputData = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    'ECartItemGroupsS' => new List<Map<String, Object>> {cartItemGroupToUpdate},
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XL,
            ccrz.ccApi.SZ_REFETCH => true
        }
    }
};

try {
```

```

    Map<String, Object> reviseCartItemGroupResults =
ccrz.ccApiCart.reviseCartItemGroup(cartItemGroupUpdateInputData);
    wasSuccessful = (Boolean) reviseCartItemGroupResults.get(ccrz.ccApi.SUCCESS);
    List<Map<String, Object>> refetchedCart = (List<Map<String, Object>)
ccrz.ccApiCart.reviseCartItemGroupResults.get(ccrz.ccApi.OBJECTLIST);
    // Do something with the fetched cart item group
} catch (Exception e) {
    // Error handling...
}

```

SEE ALSO:

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

## ccrz.ccApiCart.saveCart

Calls the `ccrz.ccApiCart.validate` method for setting the cart's `ccrz__ValidationStatus__c` field. If you're using effective accounts, this method also invokes the `ccrz.cc_hk_EffectiveAccount.saveCart` method for updating cart address information.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

## Signature

```
global static Map<String, Object> saveCart (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

[ccrz.ccLogicCartSave](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to save.

**ccrz.ccApiCart.CART\_ID**

String that specifies the ID of a cart to save.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the updated cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs

`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.MESSAGES**

`List<ccrz.cc_bean_Message>`, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiCart.CART\_OBJLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 **Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

 **Example: Example**

Save a specified cart.

```
Map<String, Object> cartToSave = new Map<String, Object>{
    ccrz.ccApiCart.CART_ID => 'CART_ID',
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
};

try {
    Map<String, Object> cartSaveResult = ccrz.ccApiCart.saveCart(cartToSave);
    Boolean wasSuccessful = cartSaveResult.get(ccrz.ccApi.SUCCESS);
    if (wasSuccessful) {
        // Execute your business logic when the save call succeeds
    }
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[ccrz.ccApiCart.validate](#)

[ccrz.cc\\_hk\\_EffectiveAccount.saveCart](#)


## ccrz.ccApiCart.setActive

Sets a single cart as the active cart for a particular owner, and updates all the owner's other open carts to inactive. Finally, this method fetches the new active cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> setActive(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccLogicCartSetActive`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### `ccrz.ccApiCart.BYEFFECTIVEACCOUNT`

Boolean

Value	Usage
<code>true</code>	Filter the cart query by the selected effective account. This method evaluates this key only when your storefront uses the <a href="#">Cart per Account mode for effective accounts</a> .
<code>false</code> (default)	Ignore effective accounts.

### `ccrz.ccApiCart.BYOWNER`

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

 **Note:**

- Sharing rules in your org can affect what data you can see when you filter by cart owner.
- If you don't include this key, this method falls back to `ccrz.cc_CallContext.currUserId`.

### `ccrz.ccApiCart.BYSTOREFRONT`

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to make active.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to make active.

## Outputs

Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.MESSAGES**

List<ccrz.cc\_bean\_Message>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is FALSE.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is false, consider rolling back the API transaction to a previous savepoint.

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where each Map<String, Object> represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 **Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see `ccrz.ccApiCart.fetch`.

 **Example: Example**

Set a specified cart to active, and parse the return data for the active cart record.

```
Map<String, Object> cartToMakeActive = new Map<String, Object>{
    ccrz.ccApiCart.CART_ID => 'CART_ID',
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
};

try {
    Map<String, Object> setActiveResults = ccrz.ccApiCart.setActive(cartToMakeActive);
```

```

    if (setActiveResults.get(ccrz.ccApiCart.CART_OBJLIST) != null) {
        List<Map<String, Object>> activeCartList = (List<Map<String, Object>>)
setActiveResults.get(ccrz.ccApiCart.CART_OBJLIST);
        String cartId = (String) activeCartList[0].get('sfid');
    }
} catch (Exception e) {
    // Error handling...
}

```

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

## ccrz.ccApiCart.transferCart

Transfers ownership of a cart to another user. By default, B2B Commerce for Visualforce calls this method when a guest user begins adding items to a cart and then registers for a storefront account. This method transfers ownership of the cart from the default owner to the newly registered user.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global static Map<String, Object> transferCart (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

[ccrz.ccLogicCartTransfer](#)

### Inputs (Required)

[Map<String, Object>](#) that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to transfer.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

**ccrz.ccApiUser.USER\_ID**

String that specifies the Salesforce ID of the user to transfer ownership to.

**ccrz.ccApiUser.USER\_NAME**

String that specifies the username of the user to transfer ownership to.

 **Important:**

- By default, this method is invoked in a context where the current user is the cart's current Owner. If the cart was originally used by a guest user, the current Owner is the [default owner for all records created by guest users](#).
- To successfully transfer ownership to another user, specify the ID or username of a user that has row-level access to the cart. Otherwise, this method doesn't return any error when ownership doesn't successfully transfer.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the updated cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs

`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


**ccrz.ccApi.MESSAGES**

`List<ccrz.cc_bean_Message>`, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.



**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where each Map<String, Object> represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

**Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

**ccrz.ccApiCart.validate**

Provides an extension point for implementing business rules that ensure that a buyer is allowed to check out from the Shopping Cart page. By default, this method calls `ccrz.ccApiCart.fetch` and returns both the fetched cart data and a Boolean flag that indicates whether the buyer can check out.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.



**Note:** This method doesn't execute when `ccrz.cc_CallContext` specifies a guest user.

## Signature

```
global static Map<String, Object> validate(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicCartValidate`



**Tip:** If you've previously extended the `ccrz.cc_api_CartExtension` class to override its `allowCheckout` or `getCartMessages` methods, migrate your customizations to an override of the `ccrz.ccLogicCartValidate.processValidate` method.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted cart ID to validate.

### `ccrz.ccApiCart.CART_ID`

String that specifies the cart ID to validate.



**Note:** Include `ccrz.ccApiCart.CART_ENCID` or `ccrz.ccApiCart.CART_ID`, but not both.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.



**Note:** By default, this method sets the following data size information:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_ASSC => false,
        ccrz.ccApi.SZ_REL => new List<String>
    }
}
```

## Outputs

`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.MESSAGES**

List<[ccrz.cc\\_bean\\_Message](#)>, where each [ccrz.cc\\_bean\\_Message](#) contains error information sent from the server to the client. Typically, these messages are included only when [ccrz.ccApi.SUCCESS](#) is `FALSE`.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiAddress.ADDRESSLIST**

List<Map<[String](#), [Object](#)>>, where each Map<[String](#), [Object](#)> represents a [ccrz\\_\\_E\\_ContactAddr\\_\\_c](#) associated with the cart's shipping or billing addresses. For more information about fetching address data with cart data, see [ccrz.ccApiCart.fetch](#).

**ccrz.ccApiCart.ALLOW\_CHECKOUT**

Boolean

Value	Usage
true	The buyer is allowed to check out from the Shopping Cart page.
false	The buyer isn't allowed to check out from the Shopping Cart page.

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<[String](#), [Object](#)>>, where each Map<[String](#), [Object](#)> represents a [ccrz\\_\\_E\\_Cart\\_\\_c](#) record that the [ccrz.ccApiCart.fetch](#) method returned.

**ccrz.ccApiProduct.PRODUCTLIST**

List<Map<[String](#), [Object](#)>>, where each Map<[String](#), [Object](#)> represents a [ccrz\\_\\_E\\_Product\\_\\_c](#) associated with the cart's items. For more information about fetching product data with cart data, see [ccrz.ccApiCart.fetch](#).

 **Example: Example**

Query a pair of cart IDs, and capture the first cart's status in a variable.

```
Map<String, Object> cartValidateQuery = new Map<String, Object>{
    ccrz.ccApiCart.CART_ID => 'CART_ID',
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
};

try {
    Map<String, Object> cartValidateData = ccrz.ccApiCart.validate(cartValidateQuery);
```

```

        Boolean allowCheckout = (Boolean).ccrz.ccUtil.defv(cartValidateData,
ccrz.ccApiCart.ALLOW_CHECKOUT, FALSE);
        List<ccrz.cc_bean_Message> cartMessages =
(List<ccrz.cc_bean_Message>)ccrz.ccUtil.defv(cartValidateData, ccrz.ccApi.MESSAGES,
new List<ccrz.cc_bean_Message>);
    } catch (Exception e) {
        // Error handling...
    }

```

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[ccrz.cc\\_api\\_CartExtension](#)

## ccrz.ccApiCategory

Query category assignments for products and data for the storefront's category taxonomy.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

IN THIS SECTION:

[fetch](#)

Executes a SOQL query of category records and returns a list of categories that match your query.

[fetchCategoryTree](#)

Generates a category tree during a batch processing job, and stores the tree in a public cache.

[getTree](#)

Returns a generated category tree from a public cache. This method populates category results on the Product List page.

[getUrlFor](#)

Returns a map where each entry associates a category's ID with its friendly URL.

## SEE ALSO:

[Salesforce Help Organize Products into Categories for B2B Commerce for Visualforce](#)


## ccrz.ccApiCategory.fetch

Executes a SOQL query of category records and returns a list of categories that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

### Service Layer Classes

#### Data Service Providers

- `ccrz.ccServiceCategory` (default)
- `ccrz.ccServiceFeatured` when `ccrz.ccApiCategory.FEATURED => 'true'` is set in the input map

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### **ccrz.ccApiCategory.CATEGORYIDLIST**

Set<String> of specific category IDs to query.

```
new Set<String>{'Category_ID_1', 'Category_ID_2'}
```

### **ccrz.ccApiCategory.CATEGORYLOCALE**

String that specifies a locale that you want to return localized category values for in the `ccrz.ccApiCategory.CATEGORYLIST` return key. The service layer translates the values from [ccrz\\_\\_E\\_CategoryI18N\\_\\_c records](#) and handles fallback languages, if necessary.

String that specifies one of your [storefront's supported locales](#).



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

### **ccrz.ccApiCategory.CATEGORYNAME**

String of a specific category name to query.

### **ccrz.ccApiCategory.FEATURED**

Boolean

Value	Usage
<i>true</i>	Instead of category data, return the storefront's featured products in <code>ccrz.ccApiCategory.CATEGORYLIST</code> . The method chains a call to the <a href="#">ccrz.ccApiProduct.fetch</a> method to query product data for the featured products.
<i>false</i> (default)	Return category data in <code>ccrz.ccApiCategory.CATEGORYLIST</code> .

### **ccrz.ccApiCategory.PARENTCATEGORY**

String of a specific category's ID whose subcategories you want to query.

## Outputs

Map<String, Object> that can include the following keys:


### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

### **ccrz.ccApi.SUCCESS**


Boolean

Value	Usage
<i>true</i>	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiCategory.CATEGORYLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [category record](#).

 **Tip:** To specify which fields the API returns for each category record, [request a specific data size](#) for the `ccrz.ccApiCategory.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields are available with different data sizes, see the [ccrz\\_\\_E\\_Category\\_\\_c](#) reference.

When you set `ccrz.ccApiCategory.FEATURED => true` in the input map, each Map<String, Object> represents a [featured product record](#) instead of a category.

 **Note:** Each featured product record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.


**ccrz.ccApiCategory.PRODUCTIDTOSKU**

Map<String, Object> that maps product IDs to product SKUs. This key is returned only when you request a small data size and specifically request the Product Categories related list:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_REL => new List<String>{'Product_Categories__r'}
    }
}
```

**ccrz.ccApiProduct.PRODUCTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [product record](#) associated with a returned category or featured product. For more information about fetching product data, see [ccrz.ccApiProduct.fetch](#).

 **Tip:** To specify which fields the API returns for each product, [request a specific data size](#) for the `ccrz.ccApiProduct.ENTITYNAME` key in your input data. You can request unique sizes for category data and product data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    },
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

```
    }
}
```

You can also exclude the product records from the return data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
        ccrz.ccApi.SZ_ASSC => false
    }
}
```

For a featured product query, set this flag under the `'featuredProducts'` key, which serves as a hard-coded entity name for featured products data.

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    'featuredProducts' => new Map<String, Object> {
        ccrz.ccApi.SZ_ASSC => false
    }
}
```

### Example: Examples

Fetch category details for a pair of category IDs, and parse the returned list for the first category's name.

```
Map<String, Object> categoryQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiCategory.CATEGORYIDLIST => new Set<String>{'Category_ID_1',
    'Category_ID_2'}
};

try {
    Map<String, Object> categoryReturnData = ccrz.ccApiCategory.fetch(categoryQuery);

    if (categoryReturnData.get(ccrz.ccApiCategory.CATEGORYLIST) != null) {
        List<Map<String, Object>> outputCategoryList = (List<Map<String, Object>>)
categoryReturnData.get(ccrz.ccApiCategory.CATEGORYLIST);
        String categoryName = (String) outputCategoryList[0].get('sfdcName');
    }
} catch (Exception e) {
    // Error handling...
}
```

Return your storefront's featured product records, and exclude all product data and related data associated with those featured products. Find the type of the first featured product that's returned.

```
Map<String, Object> featuredProductQuery = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiCategory.STOREFRONT => 'storefrontName',
    ccrz.ccApiCategory.FEATURED => true,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        'featuredProducts' => new Map<String, Object>{
            ccrz.ccApi.SZ_ASSC => false
        }
    }
};
```



```

try {
    Map<String, Object> featuredProductReturnData =
ccrz.ccApiCategory.fetch(featuredProductQuery);
    if (featuredProductReturnData.get(ccrz.ccApiCategory.CATEGORYLIST) != null) {
        List<Map<String, Object>> featuredProductList = (List<Map<String, Object>>)
featuredProductReturnData.get(ccrz.ccApiCategory.CATEGORYLIST);
        String featuredProductType = (String) featuredProductList[0].get('type');
    }
} catch (Exception e) {
    // Error handling...
}

```

## Platform Cache

When `ccrz.ccApi.SZ_ASSC => false` is set in the input map, this method checks for category data previously queried and stored in Platform Cache. This design helps improve performance on your storefront when displaying your category data.

The underlying `ccrz.ccServiceCategory.fetch` method that supports the `ccrz.ccApiCategory.fetch` method checks the Platform Cache for category data.

1. Before the `ccrz.ccApiCategory.fetch` method executes the query for category data, the underlying `ccrz.ccServiceCategory.fetch` data service method calls the `ccrz.ccService.prepareCacheKey` method.
2. If `ccrz.ccService.prepareCacheKey` finds a match in Platform Cache for the queried data, it passes the matching data to the `ccrz.ccService.prepReturn` method, which sends the data to the `ccrz.ccApiCategory.fetch` query.

If no matching data is found in the Platform Cache, the `ccrz.ccApiCategory.fetch` query executes and returns data from your org's database. In this case, the `ccrz.ccService.prepareCacheData` method is called to store the category data in Platform Cache.

To specify how long B2B Commerce for Visualforce keeps category data cached, [change the category data time to live \(TTL\) values in your storefront configuration settings](#).


## ccrz.ccApiCategory.fetchCategoryTree

Generates a category tree during a batch processing job, and stores the tree in a public cache.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetchCategoryTree (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

```
ccrz.ccServiceCategory
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.


### `ccrz.ccApi.SIZING`

`Map<String, Object>` that specifies [how much data to return from the method call](#). For this method, this key is required, and you must include `ccrz.ccApi.SZ_REL` for specifically requesting category and localized category records:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L,
        ccrz.ccApi.SZ_REL => new List<String>{'Categories__r', 'CategoryI18Ns__r'}
    }
}
```

### `ccrz.ccApiCategory.ROOTCATEGORYID`

String that specifies the ID of the storefront root category. This value corresponds to the `ccrz__CategoryID__c` field on the CC Category object.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiCategory.CATEGORYLOCALE`

String that specifies a locale that you want to generate a localized category tree for. The service layer translates the values from [ccrz\\_\\_E\\_CategoryI18N\\_\\_c records](#) and handles fallback languages, if necessary.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

### `ccrz.ccApiCategory.STOREFRONT`

String that specifies the name of the storefront to generate a category tree for. The method tries to use friendly URL values for the categories included in the tree.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiCategory.CATEGORYTREE`

`List<Map<String, Object>>` of the generated category tree.

### Example: Example

Return a category tree for a specified root category ID

```
Map<String, Object> inputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiCategory.ROOTCATEGORYID => 'root_category_ID',
    ccrz.ccApiCategory.CATEGORYLOCALE => 'en_US',
    ccrz.ccApiCategory.STOREFRONT => 'DefaultStore',
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiCategory.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L,
            ccrz.ccApi.SZ_REL => new List<String>{'Categories__r', 'CategoryI18Ns__r'}
        }
    }
};

try {
    Map<String, Object> outputData = ccrz.ccApiCategory.fetchCategoryTree(inputData);

    if (outputData.get(ccrz.ccApiCategory.CATEGORYTREE) != null) {
        // The cast to List<Map<String, Object>> is necessary...
        List<Map<String, Object>> categoryTree = (List<Map<String, Object>>)
outputData.get(ccrz.ccApiCategory.CATEGORYTREE);
    }
} catch (Exception e) {
    // Error handling...
}
```


## ccrz.ccApiCategory.getTree

Returns a generated category tree from a public cache. This method populates category results on the Product List page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> getTree(Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider


`ccrz.ccLogicCategoryGetTree`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

#### `ccrz.ccApiCategory.CATEGORYLOCALE`

String that specifies a supported locale to return the category tree for.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccApiCategory.STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**Outputs**


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiCategory.CATEGORYTREE**

List<Object> of the returned category tree.

 **Example: Example**

Return the category tree for the storefront and locale specified by `ccrz.cc_CallContext`.

```
try{
    Map<String, Object> categoryTreeToReturn = ccrz.ccApiCategory.getTree(new Map<String,
    Object>{
        ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    });

    // Get the category tree object from the return data
    List<Object> treeData = categoryTreeToReturn.get(ccrz.ccApiCategory.CATEGORYTREE);

} catch (Exception e) {
    // Error handling
}
```

SEE ALSO:

[Salesforce Help Show Subcategory Details on the Product List Page](#)


**ccrz.ccApiCategory.getUrlFor**

Returns a map where each entry associates a category's ID with its friendly URL.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> getUrlFor(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccLogicCategoryUrlFor`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiCategory.CATEGORYIDLIST`

`List<String>` of Salesforce IDs for categories whose URLs you want to return.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiCategory.CATEGORYLOCALE`

String that specifies one of your [storefront's supported locales](#).

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccApiCategory.CATEGORYLOCALES**

Set<String> of your storefront's supported locales.

**ccrz.ccApiCategory.STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiCategory.CATEGORYURLMAP**

Map<String, Object> for a single locale, where each key is a category ID and each value is the category's friendly URL.

**ccrz.ccApiCategory.LOCALECATEGORYURLMAP**

Map<String, Map<String, Object>> for multiple locales, where:

- The outer map is keyed by category ID.
- Each inner map is keyed by locale.
- Each inner map's value is the category's friendly URL for the locale.

**ccrz.ccApiCategory.LOCALESITEINDICES**

Map<String, Map<String, Object>> for multiple locales, where:

- The outer map is keyed by category ID.
- Each inner map is keyed by locale.
- Each inner map's value is the category's corresponding `ccrz__E_SiteIndex__c` record for the locale.

**ccrz.ccApiCategory.SITEINDICES**

Map<String, Object> for a single locale, where each key is a category ID and each value is the corresponding `ccrz__E_SiteIndex__c` record.

### Example: Example

Return the site indexes for a pair of category IDs.

```
try{
    // Pass category IDs to the ccrz.ccApiCategory.getUrlFor method
    Map<String, Object> categoryURLsToReturn = ccrz.ccApiCategory.getUrlFor(new
Map<String, Object>{
        ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
        ccrz.ccApiCategory.CATEGORYIDLIST => new List<String>{'Category_ID_1',
'Category_ID_2' }
    });

    // Get the map of site indexes from the return data
    Map<String, Object> categorySiteIndices =
categoryURLsToReturn.get(ccrz.ccApiCategory.SITEINDICES);

}catch(Exception e){
    // Error handling
}
```

## ccrz.ccApiConfig

Query storefront configuration data, or load new JSON-formatted configuration data.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

IN THIS SECTION:

#### [fetch](#)

Executes a SOQL query of storefront configuration data and returns the matching content as Apex or JSON strings.

#### [setBulk](#)

Loan storefront configuration data by passing in serialized JSON data or loading JSON-formatted data from a static resource. This method only adds new configuration data, and doesn't update existing configurations.




## ccrz.ccApiConfig.fetch

Executes a SOQL query of storefront configuration data and returns the matching content as Apex or JSON strings.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

### Service Layer Classes


This method doesn't support service layer extensions or overrides.

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

#### **ccrz.ccApiConfig.CONFIG\_MODULE**

String that specifies the name of a specific module whose configuration data you want to return. Use the module's API name, such as `ct` for the Category Tree module or `p1` for the Product List module.

```
ccrz.ccApiConfig.CONFIG_MODULE => 'ct'
```

**ccrz.ccApiConfig.EXTERNALLY\_SAFE**

Boolean

Value	Usage
<i>true</i>	Return configuration data that have the <a href="#">Externally Safe metadata field enabled</a> .
<i>false</i> (default)	Return all configuration data that match your query.

**ccrz.ccApiConfig.PAGE\_NAME**

String that specifies the name of a page. The method returns only the configuration data that exist for that page. Use the page keys defined in [CC Admin global settings](#), such as *hp* for Home page and *p1p* for Product List page.

```
ccrz.ccApiConfig.PAGE_NAME => 'hp'
```

 **Note:** If you don't include this key, the method returns configuration data for all pages.

**ccrz.ccApiConfig.RETURN\_SERIAL**

Boolean

Value	Usage
<i>true</i>	Return configuration data as a serialized JSON string.
<i>false</i> (default)	Return configuration data as an Apex <code>List&lt;Map&lt;String, String&gt;&gt;</code> .

**ccrz.ccApiConfig.STORE\_NAME**

String that specifies the name of a storefront.

 **Note:** If you don't include this key, the method returns configuration data that are global and apply to all storefronts.

**ccrz.ccApiConfig.USE\_CACHE**

Boolean

Value	Usage
<i>true</i> (default)	Return configuration data cached in records of the <code>ccrz__E_PublicCache__c</code> and <code>ccrz__E_PrivateCache__c</code> objects.
<i>false</i>	Return configuration data directly from records of the <code>ccrz__E_Configuration__c</code> object.

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

### ccrz.ccApiConfig.CONFIGURATIONS

List<Map<String, String>>, where each Map<String, String> represents a [configuration record](#). The key value for each configuration record returned is `moduleAPIName.configurationAPIName`, such as `ct.enabled`.

 **Tip:** To return serialized JSON instead of Map<String, String>, include `ccrz.ccApiConfig.RETURN_SERIAL => true` in the input map.

### Example: Example

Query configurations cached for the Home page in the DefaultStore and that have the Externally Safe metadata field enabled.

```
Map<String, Object> homePageConfigQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiConfig.EXTERNALLY_SAFE => true,
    ccrz.ccApiConfig.USE_CACHE => true,
    ccrz.ccApiConfig.STORE_NAME => 'DefaultStore',
    ccrz.ccApiConfig.PAGE_NAME => 'hp'
};

try {
    Map<String, Object> homePageConfigs = ccrz.ccApiConfig.fetch(homePageConfigQuery);

    Map<String, String> homePageConfigValues =
    (Map<String, String>) homePageConfigs.get(ccrz.ccApiConfig.CONFIGURATIONS);
} catch (Exception e) {
    // Error handling...
}
```

## ccrz.ccApiConfig.setBulk


Loan storefront configuration data by passing in serialized JSON data or loading JSON-formatted data from a static resource. This method only adds new configuration data, and doesn't update existing configurations.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes


This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiConfig.CONFIG_JSON`

String of the serialized JSON configuration data to load.

### `ccrz.ccApiConfig.FILE_NAME_LIST`

`List<String>` of filenames within the static resource that contain the JSON-formatted configuration data to load.

### `ccrz.ccApiConfig.STATIC_RESOURCE`

String that specifies the name of a static resource that contains the JSON data to load. Don't include the `.resource` extension. For example, if your static resource name is `configDataToLoad.resource`, specify only `configDataToLoad` for this key value.

```
ccrz.ccApiConfig.STATIC_RESOURCE => 'configDataToLoad'
```

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

 **Example: Example for a Static Resource**

Load configuration data from a static resource named `configDataToLoad.resource`, which includes two files with JSON-formatted configuration data, `configDataFile.json` and `configDataFile2.json`.

```
Map<String, Object> configDataFromStaticResource = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiConfig.STATIC_RESOURCE => 'configDataToLoad',
    ccrz.ccApiConfig.FILE_NAME_LIST => new List<String>{'configDataFile.json',
    'configDataFile2.json'};
};

try {
    ccrz.ccApiConfig.setBulk(configDataFromStaticResource);
} catch (Exception e) {
    // Error handling...
}
```

In each JSON file in the static resource, use the following format.

```
{
  "ConfigurationMetas": [
    {
      "Name": "piname",
      "APIName": "piname",
      "Description": "Header Include End Page Include",
      "ExternallySafe": false,
      "Configurations": [
        {
          "TextValue": "cc_EmptyPage",
          "Storefront": "Global",
          "PageName": "all",
          "Name": "a0M360000015nqH"
        }
      ]
    },
    {
      "Name": "enabled",
```

```

        "APIName": "enabled",
        "Description": "Header Include End Enablement flag",
        "ExternallySafe": false,
        "Configurations": [
            {
                "TextValue": "FALSE",
                "Storefront": "Global",
                "PageName": "all",
                "Name": "a0M360000015nqT"
            }
        ]
    },
    "APIName": "hie",
    "Name": "Header Include End"
}

```

### Example: Example for Serialized JSON

Load configuration data from a serialized JSON string.

 **Note:** Enclose the JSON in a set of square brackets, [ ].

```

Map<String, Object> configDataFromString = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiConfig.CONFIG_JSON => '[
        {
            "ConfigurationMetas": [
                {
                    "Name": "piname",
                    "APIName": "piname",
                    "Description": "Header Include End Page Include",
                    "ExternallySafe": false,
                    "Configurations": [
                        {
                            "TextValue": "cc_EmptyPage",
                            "Storefront": "Global",
                            "PageName": "all",
                            "Name": "a0M360000015nqH"
                        }
                    ]
                }
            ],
            {
                "Name": "enabled",
                "APIName": "enabled",
                "Description": "Header Include End Enablement flag",
                "ExternallySafe": false,
                "Configurations": [
                    {
                        "TextValue": "FALSE",
                        "Storefront": "Global",
                        "PageName": "all",
                        "Name": "a0M360000015nqT"
                    }
                ]
            }
        ]
    ]
}

```

```

        ]
      }
    ],
    "APIName": "hie",
    "Name": "Header Include End"
  }
]';

try {
  ccrz.ccApiConfig.setBulk(configDataFromString);
} catch (Exception e) {
  // Error handling...
}

```

## ccrz.ccApiContact

Query contact records associated with your B2B Commerce for Visualforce storefront users.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

IN THIS SECTION:

[fetch](#)

Executes a SOQL query of contact records and returns a list of contacts that match your query.


### ccrz.ccApiContact.fetch

Executes a SOQL query of contact records and returns a list of contacts that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

```
ccrz.ccServiceContact
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiContact.ACCOUNTID`

String that specifies the ID of an [account](#) whose associated contact record you want to query.

### `ccrz.ccApiContact.ID`

String that specifies the ID of a contact to query.

### `ccrz.ccApiContact.IDS`

`Set<String>` of specific contact IDs to query:

```
new Set<String>('Contact_ID_1', 'Contact_ID_2')
```



## Outputs


Map<String, Object> that can include the following keys:

### ccrz.ccApi.API\_VERSION

Integer that indicates which API version was used for the query.

### ccrz.ccApi.SUCCESS

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### ccrz.ccApiContact.CONTACTS

List<Map<String, Object>>, where each Map<String, Object> represents a [contact record](#).

For each contact record included `ccrz.ccApiContact.CONTACTS`, the method returns the following standard contact fields:

API Return Name	API Return Data Type
accountId	String
email	String
firstName	String
lastName	String
phone	String
salutation	String

### Example: Example

Query a set of contact record IDs, and capture each contact's email address in a string variable.

```
Map<String, Object> contactFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiContact.IDS => new
Set<String>{'Contact_ID_1', 'Contact_ID_2', 'Contact_ID_3'}
}

try {
    Map<String, Object> contactFetchData = ccrz.ccApiContact.fetch(contactFetchQuery);

    if (contactFetchData.get(ccrz.ccApiContact.CONTACTS) != null) {
```

```

        List<Map<String, Object>> contactList = (List<Map<String, Object>>)
contactFetchData.get(ccrz.ccApiContact.CONTACTS);
        for(Map<String, Object> contact : contactList){
            String contactEmail = (String)contact.get('email');
        }
    }
} catch (Exception e) {
    // Error handling...
}

```

## ccrz.ccApiCoupon

Query or clone coupon records, and programmatically apply or remove coupons from a shopping cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

#### IN THIS SECTION:

##### [cloneCoupon](#)

Clones an existing coupon, and sets default values for some of the cloned coupon's fields.

##### [fetch](#)

Executes a SOQL query of coupon records and returns a list of coupons that match your query.

##### [apply](#)

Verifies that a specific coupon code is valid for a specific shopping cart, and calculates the coupon's specified discount. The method executes in two steps. First, the method *adds* a coupon code to a cart that doesn't already have a coupon applied, and creates the `ccrz__E_CartCoupon__c` junction object. Then, the method *applies* the added coupon by calculating discounts and verifying that the coupon criteria remains satisfied, even as cart items, quantities, and prices change.

##### [remove](#)

Tries to remove coupons that are already applied to a specified shopping cart.

[updateOnUse](#)

Updates coupons that have been applied to a cart.

SEE ALSO:

[Offer Coupons for Discounts on Storefront Purchases](#)


## ccrz.ccApiCoupon.cloneCoupon

Clones an existing coupon, and sets default values for some of the cloned coupon's fields.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> cloneCoupon (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

`ccrz.ccLogicCouponClone`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCoupon.fetch` for returning the updated coupon.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCoupon.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

### **ccrz.ccApiCoupon.CARTID**

String of a specific cart ID whose applied coupon you want to clone.

### **ccrz.ccApiCoupon.CODES**

Set<String> of specific coupon codes. This method uses only the first code in the set for creating the clone.

### **ccrz.ccApiCoupon.ID**

String of a specific coupon ID to clone.

## Outputs


Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### **ccrz.ccApiCoupon.COUPONLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the new [coupon record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

### **ccrz.ccApiCoupon.ID**

ID of the clone record. This method sets the following default values for the clone's fields:

Field	Default Value
Coupon Name	<i>originalCouponName-copy</i> , such as <code>15OFF-copy</code>

Field	Default Value
Coupon Code	<code>coupon-randomInteger</code> , such as <code>coupon-12345</code>
Total Used	0

### Example: Example

Clone a coupon using its ID, and refetch the clone's record details.

```
Map<String, Object> couponToClone = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.Coupon.ID => 'Coupon_ID',
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApi.Coupon.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_REFETCH => true
        }
    }
};

try {
    Map<String, Object> clonedCoupons = ccrz.ccApi.Coupon.cloneCoupon(couponToClone);
    if (clonedCoupons.get(ccrz.ccApi.Coupon.ID) != null) {
        List<Object> clonedCoupons = (List<Object>
clonedCoupons.get(ccrz.ccApi.Coupon.COUPONLIST);
        Map<String, Object> clonedCouponDetails = (Map<String, Object>) clonedCoupons[0];
    }
} catch (Exception e) {
    // Error handling...
}
```


## ccrz.ccApiCoupon.fetch

Executes a SOQL query of coupon records and returns a list of coupons that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider


ccrz.ccServiceCoupon

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### `ccrz.ccApiCoupon.CARTIDS`

`Set<String>` of specific cart IDs whose applied coupons you want to query. The method evaluates records of the [cart coupon junction object](#) to return the coupon data for the specified carts.

```
new Set<String>{ 'Cart_ID_1', 'Cart_ID_2' }
```

### `ccrz.ccApiCoupon.CODES`

`Set<String>` of specific coupon codes to query.

```
new Set<String>{ 'Coupon_Code_1', 'Coupon_Code_2' }
```

### `ccrz.ccApiCoupon.ID`

String of a specific coupon ID to query.

### `ccrz.ccApiCoupon.IDS`

`Set<String>` of specific coupon IDs to query.

```
new Set<String>{ 'Coupon_ID_1', 'Coupon_ID_2' }
```

### `ccrz.ccApiCoupon.ISDATEFILTER`

Boolean

Value	Usage
<i>true</i> (default)	Return only coupon records where the Start Date and End Date range include the current date.
<i>false</i>	Ignore the Start Date and End Date range when querying coupon records.

**ccrz.ccApiCoupon.ISENBLED**

Boolean

Value	Usage
<i>true</i> (default)	Return only coupon records where the Enabled field is selected.
<i>false</i>	Ignore the value of the Enabled field when querying coupon records.

**ccrz.ccApiCoupon.STOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

 **Note:** This method evaluates the ID keys in the following order of precedence:

1. `ccrz.ccApiCoupon.ID`
2. `ccrz.ccApiCoupon.IDS`
3. `ccrz.ccApiCoupon.CODES`

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The call completed.
<i>false</i>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, consider rolling back the API transaction to a previous savepoint.</p>

**ccrz.ccApiCoupon.COUPONLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a coupon record. By default, this list sorts by each record's last modified date, in descending order.

**Note:**

- Each coupon record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.
- If any of the returned coupons have been applied to a cart, this list also includes the `ECartCouponsS` field, which includes [cart coupon junction object data](#).

## ccrz.ccApiCoupon.apply

Verifies that a specific coupon code is valid for a specific shopping cart, and calculates the coupon's specified discount. The method executes in two steps. First, the method *adds* a coupon code to a cart that doesn't already have a coupon applied, and creates the `ccrz__E_CartCoupon__c` junction object. Then, the method *applies* the added coupon by calculating discounts and verifying that the coupon criteria remains satisfied, even as cart items, quantities, and prices change.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global static Map<String, Object> apply(Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

`ccrz.ccLogicCouponApply`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

#### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the cart that you want to add or apply a coupon to.



**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart that you want to add or apply a coupon to.

**Inputs (Optional)**

The input map can also include the following keys:

**ccrz.ccApiCoupon.CART\_HEADER**

`ccrz__E_Cart__c` record that you want to add or apply a coupon to.

**ccrz.ccApiCoupon.CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart items for the `ccrz.ccApiCoupon.CART_HEADER`, if specified.


**ccrz.ccApiCoupon.CODE**

String that specifies a coupon code to add to the cart.

**ccrz.ccApiCoupon.SKIP\_APPLY\_CALC**

Boolean

Value	Usage
<code>true</code>	Skip the <code>ccrz.ccLogicCouponApply.applyCoupons</code> method in the logic service provider. This method adds the coupon to the cart, but doesn't apply the coupon's specified discount.
<code>false</code> (default)	Execute the <code>ccrz.ccLogicCouponApply.applyCoupons</code> method to apply the coupon's discount after adding the coupon to the cart.

 **Tip:** If you specify either `ccrz.ccApiCoupon.CART_HEADER` or `ccrz.ccApiCoupon.CART_ITEMS`, make sure to include the other key, as well. When these keys are specified, the logic chain already has the required cart data. The `ccrz.ccLogicCouponApply` logic service provider skips the `retrieveCartData` method. Otherwise, the `retrieveCartData` executes, and chains a call to the `ccrz.ccApiCart.fetch` method.

**Outputs**


Map<`String`, `Object`> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### Example: Examples

Add and apply a coupon to a cart whose encrypted ID you specify.

```
Boolean wasSuccessful = false;

Map<String, Object> couponToApplyToCart = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiCart.CART_ENCID => 'Cart_Encrypted_ID',
    ccrz.ccApiCoupon.CODE => 'someCouponCode'
};

try {
    Map<String, Object> applyCouponResult = ccrz.ccApiCoupon.apply(couponToApplyToCart);

    wasSuccessful = (Boolean)applyCouponResult.get(ccrz.ccApi.SUCCESS);
} catch (Exception e) {
    // Error handling...
}
```

Add a coupon to a cart whose encrypted ID you specify, but skip the logic that determines whether an added coupon can apply to the cart.

```
Boolean wasSuccessful = false;

Map<String, Object> couponToAddToCart = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiCart.CART_ENCID => 'Cart_Encrypted_ID',
    ccrz.ccApiCoupon.CODE => 'someCouponCode',
    ccrz.ccApiCoupon.SKIP_APPLY_CALC => true
};

try {
    Map<String, Object> addCouponResult = ccrz.ccApiCoupon.apply(couponToAddToCart);
    wasSuccessful = (Boolean)addCouponResult.get(ccrz.ccApi.SUCCESS);
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccApiCoupon.remove

Tries to remove coupons that are already applied to a specified shopping cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global static Map<String, Object> remove (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicCouponRemove`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the cart to remove coupons from.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to remove coupons from.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiCoupon.REPRICE_CART`

Boolean

Value	Usage
<code>true</code>	Chain a call to the <code>ccrz.ccApiCart.price</code> method to reprice the cart after removing the coupons. <b>Warning:</b> This chained call isn't safe for Apex callouts.
<code>false</code> (default)	Don't chain a call to the <code>ccrz.ccApiCart.price</code> method.

## Outputs


Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is false, consider rolling back the API transaction to a previous savepoint.

### Example: Example

Remove coupons from a specific cart.

```
Boolean wasSuccessful = false;

Map<String, Object> cartToRemoveCouponsFrom = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiCart.CART_ID => 'Cart_ID'
};

try {
    Map<String, Object> removeCouponResult =
ccrz.ccApiCoupon.remove(cartToRemoveCouponsFrom);
    wasSuccessful = (Boolean)removeCouponResult.get(ccrz.ccApi.SUCCESS);
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[ccrz.ccApiCart.price](#)

## ccrz.ccApiCoupon.updateOnUse

Updates coupons that have been applied to a cart.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global static Map<String, Object> updateOnUse (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicCouponUpdateUse`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiCoupon.INCREMENT_AMOUNT`

A positive or negative integer to add to the `ccrz__TotalUsed__c` field of the specified coupons.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

Include one of the following keys. If none of these keys are included, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiCoupon.CODE`

String that specifies the value of the `ccrz__CouponCode__c` field of the coupon to update.

### `ccrz.ccApiCoupon.CODES`

`List<String>` that specifies `ccrz__CouponCode__c` field values of the coupons to update.

### `ccrz.ccApiCoupon.ID`

String that specifies the Salesforce ID of the coupon to update.

### `ccrz.ccApiCoupon.IDS`

`List<String>` that specifies the Salesforce IDs of coupons to update.



**Note:** This method checks for these keys in the following order of precedence:

1. `ccrz.ccApiCoupon.ID`
2. `ccrz.ccApiCoupon.IDS`
3. `ccrz.ccApiCoupon.CODE`
4. `ccrz.ccApiCoupon.CODES`

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### **ccrz.ccApiCoupon.LOCK\_COUPON**

Boolean

Value	Usage
<i>true</i> (default)	Query the specified coupons using the SOQL <code>FOR UPDATE</code> keyword, which locks the coupons from other changes until the update completes.
<i>false</i>	Don't lock coupons during this method's execution.

## Outputs


Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The call completed.
<i>false</i>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### **ccrz.ccApiCoupon.COUPONS\_UPDATED**

List<[ccrz\\_\\_E\\_Coupon\\_\\_c](#)> of the updated coupon records.

SEE ALSO:

[ccrz\\_\\_E\\_Coupon\\_\\_c](#)

[FOR UPDATE](#)

## ccrz.ccApi18N

Query locales and locale-specific resources for internationalizing a storefront.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

IN THIS SECTION:

### [fetchPageLabels](#)

Executes a SOQL query of page label records and returns page label data for a specific locale. This method doesn't perform any DML operations.

### [getAvailableLocales](#)

References the storefront settings and returns the list of the storefront's enabled locales.

### [getBrowserLocales](#)

Checks the `Accept-Language` request HTTP header to determine which locales the buyer's browser supports.

### [getLocale](#)

Returns the current buyer's locale.

### [setLocale](#)

Sets a new locale for the current authenticated buyer. Before setting the new locale, this method verifies that the specified locale is valid and included in the list of supported storefront locales.

## ccrz.ccApi18N.fetchPageLabels


Executes a SOQL query of page label records and returns page label data for a specific locale. This method doesn't perform any DML operations.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetchPageLabels (Map<String, Object>)
```

## Service Layer Classes

This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.


## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiI18N.LOCALE`

String that specifies one of your [storefront's supported locales](#).

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc.CallContext.userLocale`.

 **Note:** If a page label isn't available for a specified locale, the method [falls back to a more generic locale or the storefront's default locale](#).

### `ccrz.ccApiI18N.PAGE_LABEL_NAMES`

`Set<String>` of the names of specific page label records to query. Match the value of the Page Label Name field on a page label record.

```
new Set<String>{ 'Page_Label_Name_1', 'Page_Label_Name_2' }
```

### `ccrz.ccApiI18N.PAGE_NAME`

String that specifies the page key for a particular storefront page whose labels you want to query, such as *HP* or *PLP*.

### `ccrz.ccApiI18N.RETURN_SERIAL`

Boolean




Value	Usage
<code>true</code>	Return page label data as serialized JSON.
<code>false</code> (default)	Return page label data as a <code>Map&lt;String, String&gt;</code> .

**ccrz.ccApiI18N.STORE\_NAME**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

 **Note:** If the method can't determine a storefront, the method returns only the page labels where the value of the Storefront field is `Global`.

**ccrz.ccApiI18N.USE\_CACHE**

Boolean

Value	Usage
<code>true</code>	Query page label data from the B2B Commerce for Visualforce public cache. If the specific page label data isn't available in the cache, the method queries <code>ccrz__E_PageLabel__c</code> object data.
<code>false</code> (default)	Query page label data directly from <code>ccrz__E_PageLabel__c</code> records.

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiI18N.PAGE\_LABELS**

`Map<String, String>` keyed by page label names, where each value is the localized page label text.

 **Note:** If you include `ccrz.ccApiI18N.RETURN_SERIAL => true` in the input map, this data is formatted as serialized JSON.

### Example: Example

Query all page labels localized into Japanese for the Home page from the public cache. From the returned page label data, capture the value of the Header\_MyAccount page label.

```
Map<String, Object> pageLabelsToFetch = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi18N.LOCALE => 'ja_JP',
    ccrz.ccApi18N.STORE_NAME => 'DefaultStore',
    ccrz.ccApi18N.PAGE_NAME => 'HP',
    ccrz.ccApi18N.USE_CACHE => true
};

Map<String, Object> pageLabelsFetched =
ccrz.ccApi18N.fetchPageLabels(pageLabelsToFetch);

System.debug('pageLabelsFetched KEYS=' +
JSON.serializePretty(pageLabelsFetched.keySet()));
Boolean success = (Boolean)pageLabelsFetched.get(ccrz.ccApi.SUCCESS);

if(success) {
    Map<String, Object> pageLabels = (Map<String,
Object>)pageLabelsFetched.get(ccrz.ccApi18N.PAGE_LABELS);
    System.debug('pageLabels=' + JSON.serializePretty(pageLabels));

    String headerMyAccountLabel = (String)pageLabels.get('Header_MyAccount');
    System.debug('headerMyAccountLabel=' + JSON.serializePretty(headerMyAccountLabel));
} else {
    List<ccrz.cc_bean_Message> messages =
(List<ccrz.cc_bean_Message>)pageLabelsFetched.get(ccrz.ccApi.MESSAGES);
    System.debug('messages=' + messages);
}
```

SEE ALSO:

[ccrz\\_\\_E\\_PageLabel\\_\\_c](#)

[ccrz\\_\\_E\\_PageLabel18N\\_\\_c](#)

[Salesforce Help Translate Storefront Text for Buyers in Different Locales](#)

[Salesforce Help Customize Storefront Headings and Text with Page Labels](#)

## ccrz.ccApi18N.getAvailableLocales


References the storefront settings and returns the list of the storefront's enabled locales.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> getAvailableLocales (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


```
ccrz.ccLogicI18NGetAvailableLocales
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiI18N.PARAM_STOREFRONT`

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.


## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApi18N.PARAM\_AVAILABLE\_LOCALES**

Set<String> of the storefront's enabled locales.

SEE ALSO:

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)


**ccrz.ccApi18N.getBrowserLocales**

Checks the `Accept-Language` request HTTP header to determine which locales the buyer's browser supports.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

**Signature**

```
global static Map<String, Object> getBrowserLocales (Map<String, Object>)
```

**Service Layer Classes****Logic Service Provider**

[ccrz.ccLogic18NGetBrowserLocales](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi18N.PARAM_BROWSER_LOCALES`

`Set<String>` of the browser's supported locales.

SEE ALSO:

[Mozilla Developers Accept-Language](#)


## `ccrz.ccApi18N.getLocale`

Returns the current buyer's locale.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> getLocale(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccLogicI18NGetLocale`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

#### **ccrz.ccApiI18N.PARAM\_STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

`Map<String, Object>` that can include the following keys:

#### **ccrz.ccApiI18N.PARAM\_LOCALE**

String that specifies the current buyer's locale.


## ccrz.ccApi18N.setLocale

Sets a new locale for the current authenticated buyer. Before setting the new locale, this method verifies that the specified locale is valid and included in the list of supported storefront locales.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> setLocale (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccLogicI18NSetLocale`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiI18N.PARAM_LOCALE`

String that specifies one of your [storefront's supported locales](#)—such as `ja_JP`— or generic locale—such as `ja`—to set for the user.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiI18N.PARAM_STOREFRONT`

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApiI18N.PARAM_LOCALE`

String that specifies the set locale for the current buyer.

 **Note:** If the method couldn't set the locale successfully, this key's value is null.

SEE ALSO:

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

## ccrz.ccApilInvoice

Query invoice records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

IN THIS SECTION:

[fetch](#)

Executes a SOQL query of invoice records and returns a list of records that match your query.

SEE ALSO:

[Generate an Invoice for a Buyer's Order](#)

### ccrz.ccApilInvoice.fetch

Executes a SOQL query of invoice records and returns a list of records that match your query.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.




## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceInvoice`

 **Note:** To extend the default invoice functionality, use `ccrz.cc_hk_Invoice` instead of extending this API.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### `ccrz.ccApiInvoice.BYASC`

Boolean

Value	Usage
<code>true</code>	Sort returned invoices ID, Date Due value, or Date Issued value—whichever you specify—in ascending order.
<code>false</code> (default)	Sort returned invoices by specified criteria in descending order.

### `ccrz.ccApiInvoice.BYDUEDATE`

Boolean

Value	Usage
<code>true</code>	Sort returned invoices by the value of the Date Due field.
<code>false</code> (default)	Sort returned invoices by other criteria.

### `ccrz.ccApiInvoice.BYID`

Boolean

Value	Usage
<i>true</i>	Sort returned invoices by ID.
<i>false</i> (default)	Sort returned invoices by other criteria.

**ccrz.ccApiInvoice.BYISSUEDATE**

Boolean

Value	Usage
<i>true</i>	Sort returned invoices by the value of the Date Issued field.
<i>false</i> (default)	Sort returned invoices by other criteria.

**ccrz.ccApiInvoice.DUEDATE**

Date that specifies a balance due date to query invoice records for.

```
ccrz.ccApiInvoice.DUEDATE => Date.newInstance(YYYY, MM, DD);
```

**ccrz.ccApiInvoice.ID**

String that specifies the ID of an invoice to query.

**ccrz.ccApiInvoice.IDS**

Set&lt;String&gt; of specific invoice IDs you want to query:

```
new Set<String>{'Invoice_ID_1', 'Invoice_ID_2'}
```

**ccrz.ccApiInvoice.ISSUEDATE**

Date that specifies an issued date to query invoice records for.

```
ccrz.ccApiInvoice.ISSUEDATE => Date.newInstance(YYYY, MM, DD);
```

**ccrz.ccApiInvoice.ORDERIDS**

Set&lt;String&gt; of specific order IDs whose invoices you want to query:

```
new Set<String>{'Order_ID_1', 'Order_ID_2'}
```

**ccrz.ccApiInvoice.OWNERID**

String that specifies the ID of a user whose invoice records you want to query.

**ccrz.ccApiInvoice.STOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs


Map&lt;String, Object&gt; that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .


**ccrz.ccApiInvoice.INVOICES**List<Map<String, Object>>, where each Map<String, Object> represents an [invoice record](#).

 **Tip:** To specify which fields the API returns for each invoice record, [request a specific data size](#) for the `ccrz.ccApiInvoice.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields are available with different data sizes, see the [ccrz\\_\\_E\\_Invoice\\_\\_c](#) reference.

**ccrz.ccApiOrder.ORDERLIST**List<Map<String, Object>>, where each Map<String, Object> represents an order record related to an invoice in `ccrz.ccApiInvoice.INVOICES`.

 **Tip:** To specify which fields the API returns for each order record, [request a specific data size](#) for the `ccrz.ccApiOrder.ENTITYNAME` key in your input data. You can request unique sizes for cart data and product data:


```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    },
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
    }
}
```

You can also exclude the order records from the return data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L,
        ccrz.ccApi.SZ_ASSC => false
    }
}
```

**ccrz.ccApiAddress.ADDRESSLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [contact address record](#) related to an invoice in ccrz.ccApiInvoice.INVOICES.

 **Tip:** To specify which fields are returned for each address, add a ccrz.ccApiAddress.ENTITYNAME block to your input sizing:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    },
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
    },
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
    }
}
```

You can also exclude the address records from the return data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L,
        ccrz.ccApi.SZ_ASSC => false
    }
}
```

 **Example: Example**

Query for a specific invoice ID, and capture the value of the invoice's Type field in a string variable.

```
Map<String, Object> inputData = (new Map<String, Object>){
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiInvoice.STOREFRONT => 'DefaultStore',
    ccrz.ccApiInvoice.ID => 'Invoice_ID',
    ccrz.ccApi.SIZING => new Map<String, Object>{
        ccrz.ccApiInvoice.ENTITYNAME => new Map<String, Object>{
            ccrz.ccApi.SZ_DATA => ccrz.ccAPI.SZ_XL
        }
    }
});

try {
    Map<String, Object> outputData = ccrz.ccApiInvoice.fetch(inputData);
    if (outputData.get(ccrz.ccAPIInvoice.INVOICES) != null) {
        // The cast to List<Map<String, Object>> is necessary...
        List<Map<String, Object>> invoiceList = (List<Map<String, Object>>)
outputData.get(ccrz.ccApiInvoice.INVOICES);
        for(Map<String, Object> invoice : invoiceList){
            // The cast, again, is necessary...
            String invoiceType = (String)invoice.get('type');
        }
    }
} catch (Exception e) {
```

```

    // Error handling...
}

```

SEE ALSO:

[Generate an Invoice for a Buyer's Order](#)

## ccrz.ccApiMenu

Query menus and menu items.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

IN THIS SECTION:

[fetch](#)

Executes a SOQL query of menu records and their nested menu items, and returns a list of menus that match you query.

## ccrz.ccApiMenu.fetch


Executes a SOQL query of menu records and their nested menu items, and returns a list of menus that match you query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider


`ccrz.ccServiceMenu`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#).

 **Note:** Specific data size requests don't apply to this method. The same fields are always returned for each object, regardless of any specific data size request in your input map.

### `ccrz.ccApiMenu.LOCALE`

String that specifies a locale whose internationalized menu and menu item values you want to return.

### `ccrz.ccApiMenu.STOREFRONT`

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint.</a>

**ccrz.ccApiMenu.MENULIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_Menu__c` record. Any nested `ccrz__E_MenuItem__c` records are included in the `children` field.

 **Note:** Each menu or menu item record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.

 **Example: Example**

Fetch menu records for the DefaultStore and en\_US locale. Parse the return data for a menu entry's URL.

```
Map<String, Object> menuFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiMenu.STOREFRONT => 'DefaultStore',
    ccrz.ccApiMenu.LOCALE => 'en_US'
};

try {
    Map<String, Object> menuFetchData = ccrz.ccApiMenu.fetch(menuFetchQuery);
    if (menuFetchData.get(ccrz.ccApiMenu.MENULIST) != null) {
        List<Map<String, Object>> outputMenuList = (List<Map<String, Object>>)
menuFetchData.get(ccrz.ccApiMenu.MENULIST);
        String menuLinkValue = (String) outputMenuList[0].get('linkURL');
    }
} catch (Exception e) {
    // Error handling...
}
```

## ccrz.ccApiOrder

Query, search, split, reorder, or cancel order records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

#### [authCancel](#)

Calls the `ccrz.cc_api_OutboundOrderCancel` extension point class for determining whether a buyer can cancel an order.

#### [cancel](#)

Changes the status of an order to `Cancelled` after the `ccrz.ccApiOrder.authCancel` method determines that the order is eligible for cancellation.

#### [fetch](#)

Executes a SOQL query of order records, and returns a list of records that match your query.

#### [fetchOrderHistory](#)

Calls the `ccrz.ccApiOrder.search` method for searching order history for specific search terms. Then, this method calls the `ccrz.ccApiOrder.fetch` method with the order IDs that match the search results, and populates the results with more order details.

#### [fetchOrderItems](#)

Executes a SOQL query of order item records, and returns a list of order items that match your query.

#### [getOrdersToSplit](#)

Determines which orders to include in the `ccrz.ccSplitOrderJob` batch processing class. By default, this method includes all orders where the `ccrz__SplitProcessed__c` field is `FALSE`.

#### [reorder](#)

Copies the data for a completed order and its order items to a new cart and cart items.

#### [restore](#)

Reopens a cart after the corresponding order fails when attempting to place the order asynchronously. This method resets the status of the cart back to `Open` and makes that cart active. You can restore an order only one time.

#### [search](#)

Executes a SOSL query of order records, and returns the IDs of orders that match the search query.

#### [splitOrder](#)

Splits orders by seller. This method creates a separate child order for each seller referenced in the parent order. For example, let's say that a parent contains three line items—two items associated with one seller and another item associated with a different seller. This method creates two child orders, where each child order corresponds to each seller.




## ccrz.ccApiOrder.canCancel

Calls the `ccrz.cc_api_OutboundOrderCancel` extension point class for determining whether a buyer can cancel an order.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> canCancel (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

`ccrz.ccLogicOrderAuthCancel`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### `ccrz.ccApiOrder.PARAM_ORDER_STATUS`

String that specifies the status of the order to pass to the `ccrz.cc_api_OutboundOrderCancel.eligibleForCancellation` extension point method.

### Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiOrder.AUTHCANCEL**

Boolean

Value	Usage
true	<p>The buyer can cancel the order. By default, the <a href="#">ccrz.cc_api_OutboundOrderCancel.eligibleForCancellation</a> method returns <code>true</code> when this method's <code>ccrz.ccApiOrder.PARAM_ORDER_STATUS</code> input specifies one of the following values:</p> <ul style="list-style-type: none"> <li>Cancelled</li> <li>Pending Cancel</li> <li>Shipped</li> </ul>
false	The buyer can't cancel the order.

SEE ALSO:

[ccrz.cc\\_api\\_OutboundOrderCancel.eligibleForCancellation](#)

**ccrz.ccApiOrder.cancel**


Changes the status of an order to `Cancelled` after the `ccrz.ccApiOrder.authCancel` method determines that the order is eligible for cancellation.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> cancel (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccLogicOrderCancel`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.


Include one of the following keys. If none of these keys are included, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiOrder.PARAM_ORDER_ENCID`

Encrypted ID of the order to cancel.

### `ccrz.ccApiOrder.PARAM_ORDER_ENCID_LIST`

`Set<String>` that includes the encrypted ID of only one order to cancel.


 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

### `ccrz.ccApiOrder.PARAM_ORDER_ID`

Salesforce ID of the order to cancel.

### `ccrz.ccApiOrder.PARAM_ORDER_ID_LIST`

`Set<String>` that includes the Salesforce ID of only one order to cancel.

 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiOrder.fetch` for returning the canceled order.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

**Outputs**


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiOrder.ORDERLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the canceled `ccrz__E_Order__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

 **Example: Example**

Cancel an order by Salesforce ID.

```
Map<String, Object> orderCancelData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiOrder.PARAM_ORDER_ID => 'order_ID'
};

try {
    Map<String, Object> orderCancelResults = ccrz.ccApiOrder.cancel(orderCancelData);

    System.debug((Boolean) orderCancelResults.get(ccrz.ccApi.SUCCESS));
} catch (Exception e) {
    // Error handling...
}
```


## ccrz.ccApiOrder.fetch

Executes a SOQL query of order records, and returns a list of records that match you query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider


`ccrz.ccServiceOrder`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

#### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

**ccrz.ccApiOrder.PARAM\_BY\_AMOUNT**

Boolean

Value	Usage
<i>true</i>	Sort returned orders by <code>ccrz__TotalAmount__c</code> value in descending order.
<i>false</i> (default)	Sort orders by other criteria.

**ccrz.ccApiOrder.PARAM\_BY\_ASC**

Boolean

Value	Usage
<i>true</i>	Sort returned orders by specified criteria ( <code>ccrz.ccApiOrder.PARAM_BY_AMOUNT</code> , <code>ccrz.ccApiOrder.PARAM_BY_DATE</code> , <code>ccrz.ccApiOrder.PARAM_BY_ORDERNUMBER</code> , or <code>ccrz.ccApiOrder.PARAM_BY_STATUS</code> ) in ascending order.
<i>false</i> (default)	Sort orders in descending order.

**ccrz.ccApiOrder.PARAM\_BY\_DATE**

Boolean

Value	Usage
<i>true</i>	Sort returned orders by <code>ccrz__OrderDate__c</code> value in descending order.
<i>false</i> (default)	Sort orders by other criteria.

**ccrz.ccApiOrder.PARAM\_BY\_ORDERNUMBER**

Boolean

Value	Usage
<i>true</i>	Sort returned orders by <code>ccrz__OrderNumber__c</code> value in descending order.
<i>false</i> (default)	Sort orders by other criteria.

**ccrz.ccApiOrder.PARAM\_BY\_STATUS**

Boolean

Value	Usage
<i>true</i>	Sort returned orders by <code>ccrz__OrderStatus__c</code> value in descending order.
<i>false</i> (default)	Sort orders by other criteria.

**ccrz.ccApiOrder.PARAM\_FROM\_DATE**

String that specifies a locale-specific date format, such as *mm/dd/yyyy* for the en\_US locale. This method returns order records where the `ccrz__OrderDate__c` value occurs after this specified date.

**ccrz.ccApiOrder.PARAM\_ORDER\_EFFECT\_ID**

String that specifies the ID of an effective account for filtering orders.

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID\_LIST**

Set<String> of encrypted IDs for the orders to query.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID\_LIST**

Set<String> of Salesforce IDs for the orders to query.

**ccrz.ccApiOrder.PARAM\_ORDER\_STATUS**

String that specifies a `ccrz__OrderStatus__c` value for filtering orders.

**ccrz.ccApiOrder.PARAM\_OWNERID**

String that specifies the Salesforce ID of the user that owns the orders to query.

**ccrz.ccApiOrder.PARAM\_TO\_DATE**

String that specifies a locale-specific date format, such as *mm/dd/yyyy* for the en\_US locale. This method returns order records where the `ccrz__OrderDate__c` value occurs before this specified date.

**ccrz.ccApiOrder.SUBSCORDER**

String that specifies the Salesforce ID of a `ccrz__E_Subscription__c` record associated with the orders to the query. To return orders where the `ccrz__CCSubscription__c` field is blank, pass an empty string this key's value.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiAddress.ADDRESSLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_ContactAddr__c` associated with the order's shipping or billing addresses. For more information about fetching address data, see [ccrz.ccApiAddress.fetch](#).

-  **Tip:** To specify which fields the API returns for each contact address, [request a specific data size](#) for the `ccrz.ccApiAddress.ENTITYNAME` key in your input data. You can request unique sizes for order data and contact address data:


```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    },
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

You can also exclude the address records from the return data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
        ccrz.ccApi.SZ_ASSC => false
    }
}
```

#### **ccrz.ccApiOrder.ORDERLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_Order__c` record.


-  **Tip:** To specify which fields the API returns for each order record, [request a specific data size](#) for the `ccrz.ccApiOrder.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields and related lists (such as order items and order item groups) are available with different data sizes, see the `ccrz__E_Order__c` reference.

#### **ccrz.ccApiProduct.PRODUCTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_Product__c` associated with the order's items. For more information about fetching product data, see `ccrz.ccApiProduct.fetch`.


-  **Tip:** To specify which fields the API returns for each product record, [request a specific data size](#) for the `ccrz.ccApiProduct.ENTITYNAME` key in your input data. You can request unique sizes for order data and product data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    },
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```



You can also exclude the product records from the return data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
        ccrz.ccApi.SZ_ASSC => false
    }
}
```

 **Note:** The `ccrz.ccApiProduct.PRODUCTLIST` output can include products that were ordered but are no longer current on your storefront.

### Example: Example

Fetch order records for a pair of specified order IDs, and capture the first result's order status. Evaluate the list of products included in the order return data.

```
Map<String, Object> orderFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiOrder.PARAM_ORDER_ID_LIST => new Set<String>{'order_ID_1', 'order_ID_2'}
};

try {
    Map<String, Object> orderFetchResults = ccrz.ccApiOrder.fetch(orderFetchQuery);
    if (orderFetchResults.get(ccrz.ccApiOrder.ORDERLIST) != null) {
        List<Map<String, Object>> outputOrdersList = (List<Map<String, Object>>)
orderFetchResults.get(ccrz.ccApiOrder.ORDERLIST);
        String orderName = (String) outputOrdersList[0].get('orderStatus');
    }

    if (orderFetchResults.get(ccrz.ccApiProduct.PRODUCTLIST) != null) {
        List<Map<String, Object>> outputProductsList = (List<Map<String, Object>>)
orderFetchResults.get(ccrz.ccApiProduct.PRODUCTLIST);
        // Evaluate the products included in the order
    }
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[Salesforce Help Allow a Buyer to Manage Their Orders on the My Account Page](#)


## ccrz.ccApiOrder.fetchOrderHistory

Calls the `ccrz.ccApiOrder.search` method for searching order history for specific search terms. Then, this method calls the `ccrz.ccApiOrder.fetch` method with the order IDs that match the search results, and populates the results with more order details.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetchOrderHistory(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicOrderHistory`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiOrder.PARAM_ORDER_ID_LIST`

`Set<String>` of Salesforce IDs for the orders to query.

### `ccrz.ccService.SEARCHSTRING`

String to use for searching order records.

For information about how this method evaluates these parameters, see the reference for the [ccrz.ccLogicOrderHistory.fetchHistory](#) logic service provider method.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

**ccrz.ccApiOrder.PARAM\_ORDERLIMIT**

Integer that specifies the maximum number of order records to query at one time.

**ccrz.ccApiOrder.PARAM\_ORDER\_ITEM\_LIMIT**

Integer that specifies the maximum number of order item records to query at one time.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, consider rolling back the API transaction to a previous savepoint.</p>

This method's output also includes the output of `ccrz.ccApiOrder.search`, `ccrz.ccApiOrder.fetch`, and `ccrz.ccApiOrder.fetchOrderItems`, if applicable.

SEE ALSO:

[ccrz.ccApiOrder.fetch](#)

[ccrz.ccApiOrder.search](#)


## ccrz.ccApiOrder.fetchOrderItems

Executes a SOQL query of order item records, and returns a list of order items that match you query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetchOrderItems (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider


`ccrz.ccServiceOrderItem`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiOrder.PARAM_ORDER_ID_LIST`

`Set<String>` of Salesforce IDs for the orders whose items you want to query.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### `ccrz.ccApiOrder.OISERVSPLIT`

Integer that specifies the buffer size of order items to process at one time. The default value is 200.

## Outputs

`Map<String, Object>` that can include the following keys:


### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiOrder.ORDERITEMATTRIDS**


Set<String> of Salesforce IDs for attributes that correspond to the returned order items.

**ccrz.ccApiOrder.ORDERITEMPRODUCTIDS**

Set<String> of Salesforce IDs for products that correspond to the returned order items.

**ccrz.ccApiOrder.ORDERITEMSBYID**

Map<String, List<Map<String, Object>>>, keyed by parent `ccrz__E_Order__c`, where each Map<String, Object> represents a `ccrz__E_OrderItem__c` record that matches the query.

 **Tip:** To specify which fields the API returns for each cart record, [request a specific data size](#) for the `ccrz.ccApiOrder.OI_ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.OI_ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

To instead return a Map<String, List<`ccrz__E_OrderItem__c`>>, set `ccrz.ccApi.SZ_SKIPTRZ => true` in your input map:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.OI_ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_SKIPTRZ => true
    }
}
```

**ccrz.ccApiOrder.getOrdersToSplit**


Determines which orders to include in the `ccrz.ccSplitOrderJob` batch processing class. By default, this method includes all orders where the `ccrz__SplitProcessed__c` field is FALSE.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> getOrdersToSplit(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


```
ccrz.ccLogicOrderGetOrdersToSplit
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiOrder.PARAM_STOREFRONT`

String that specifies the name of the storefront to split orders for.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiOrder.SPLITSELECTOR**

[QueryLocator](#) for the [ccrz.ccSplitOrderJob](#) batch processing class to execute against.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz.ccSplitOrderJob](#)


## ccrz.ccApiOrder.reorder

Copies the data for a completed order and its order items to a new cart and cart items.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> reorder (Map<String, Object>)
```

### Service Layer Classes

**Logic Service Provider**

[ccrz.ccLogicOrderReOrder](#)

### Inputs (Required)

[Map<String, Object>](#) that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the [ccrz.ccApi.CURRENT\\_VERSION](#) constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a [ccrz.ccApi.NoApiVersionException](#).


Include one of the following keys. If none of these keys are included, the method returns a `ccrz.ccApi.MissingInputException`.

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID**

Encrypted ID of the order to copy.

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID\_LIST**

Set<String> that includes the encrypted ID of only one order to copy.


 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID**

Salesforce ID of the order to copy.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID\_LIST**

Set<String> that includes the Salesforce ID of only one order to copy.

 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the created cart.


```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs

Map<String, Object> that can include the following keys:

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the created `ccrz__E_Cart__c` record.

 **Note:** The output data includes this key only when you extend the default `ccrz.ccLogicOrderReOrder` class to reference the legacy `ccrz.cc_hk_Order.reorder` method. The managed package provides an example extension, `ccrz.ccLogicOrderReOrder7HK`, that you can use as a starting point for extending `ccrz.ccLogicOrderReOrder`.

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


**ccrz.ccApi.MESSAGES**

List<`ccrz.cc_bean_Message`>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

**ccrz.ccApi.SUCCESS**

Boolean



Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

**ccrz.ccApiAddress.ADDRESSLIST**


List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_ContactAddr__c` associated with the cart's shipping or billing addresses. For more information about fetching address data with cart data, see [ccrz.ccApiCart.fetch](#).

**ccrz.ccApiCart.CART\_ID**

Salesforce ID of the created `ccrz__E_Cart__c` record.


**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the created `ccrz__E_Cart__c` record.

 **Note:** This output is a duplicate of `ccrz.cc_hk_Order.PARAM_ENC_CART_ID`.


**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the created `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

**ccrz.ccApiProduct.PRODUCTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_Product__c` associated with the cart's items. For more information about fetching product data with cart data, see [ccrz.ccApiCart.fetch](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz.ccApiCart.fetch](#)

## ccrz.ccApiOrder.restore

Reopens a cart after the corresponding order fails when attempting to place the order asynchronously. This method resets the status of the cart back to Open and makes that cart active. You can restore an order only one time.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global static Map<String, Object> restore(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicOrderRestore`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include one of the following keys. If none of these keys are included, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiOrder.PARAM_ORDER_ENCID`

String that specifies the encrypted ID of the order to restore.

### `ccrz.ccApiOrder.PARAM_ORDER_ID`

String that specifies the Salesforce ID of the order to restore.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`


Integer that indicates which API version was used for the query.

### `ccrz.ccApi.MESSAGES`

List<`ccrz.cc_bean_Message`>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**'originatedCart'**

String that specifies the encrypted ID of the reopened cart that corresponds to the order that failed to place.

 **Example: Example**

Pass the encrypted ID of an order to restore.

```
Map<String, Object> orderToRestore = new Map<String, Object>{
    ccrz.ccApiOrder.PARAM_ORDER_ENCID => 'ORDER_ENCID',
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
};

try {
    Map<String, Object> orderRestoreResult = ccrz.ccApiOrder.restore(orderToRestore);

    List<ccrz.cc_bean_Message> orderMessages =
(List<ccrz.cc_bean_Message>)ccrz.ccUtil.defv(orderRestoreResult, ccrz.ccApi.MESSAGES,
new List<ccrz.cc_bean_Message>);
} catch (Exception e) {
    // Error handling...
}
```


## ccrz.ccApiOrder.search

Executes a SOSL query of order records, and returns the IDs of orders that match the search query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> search(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceOrder`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### `ccrz.ccService.SEARCHSTRING`

String to use for searching order records.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

### Outputs


`Map<String, Object>` that can include the following keys:

#### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

#### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

#### `ccrz.ccService.SEARCHRESULTS`

`Set<String>` of order IDs that match the search string.


## `ccrz.ccApiOrder.splitOrder`

Splits orders by seller. This method creates a separate child order for each seller referenced in the parent order. For example, let's say that a parent contains three line items—two items associated with one seller and another item associated with a different seller. This method creates two child orders, where each child order corresponds to each seller.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.


## Signature

```
global static Map<String, Object> splitOrder (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicOrderSplitOrder`

 **Note:** By default, the `ccrz.ccSplitOrderJob` batch processing class calls this method, but you can also call this method directly from subscriber code. To split orders by criteria other than sellers, extend the `ccrz.ccLogicOrderSplitOrder` logic service provider class.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiOrder.PARAM_ORDER_ID_LIST`

`Set<String>` that includes the Salesforce IDs of the orders to split.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiOrder.PARAM_STOREFRONT`

String that specifies the name of the storefront to split orders for.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Outputs

Map<[String](#), [Object](#)> that can include the following keys:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

SEE ALSO:

[ccrz.ccSplitOrderJob](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## `ccrz.ccApiPriceList`

Query price lists and price list items, which specify an account group's entitled prices for particular products.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

IN THIS SECTION:

[fetch](#)

Executes a SOQL query of price list records, and returns a list of price lists that match your query.

[fetchEntitled](#)

Executes a SOQL or SOSL query of price list records, and returns a list of price list IDs that the current account is entitled to. You can filter by account group or price group entitlements.

[fetchPriceListItems](#)

Executes a SOQL query of price list item records, and returns a list of price list item IDs.

SEE ALSO:

[Salesforce Help Manage Prices for Storefront Products with Price Lists](#)


## ccrz.ccApiPriceList.fetch

Executes a SOQL query of price list records, and returns a list of price lists that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider

`ccrz.ccServicePL`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### **ccrz.ccApiPriceList.CURRCODE**

String that specifies an ISO 4217 currency code, such as *USD* or *JPY*.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userCurrency`.

### **ccrz.ccApiPriceList.EFFACCTID**

String that specifies the ID of an [account](#) record to filter price lists by. The method returns only the price list records that are assigned to the specified account's account group.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.currAccountId`.

### **ccrz.ccApiPriceList.EFFDATE**

Date to filter price lists by. The method returns only the price list records where the date is included in the Start Date and End Date range.



**Note:** If the input data doesn't include this key, the method falls back to the current date.

### **ccrz.ccApiPriceList.IDS**

Set<String> of specific price list IDs to query:

```
new Set<String>{ 'Price_List_ID_1', 'Price_List_ID_2' }
```

### **ccrz.ccApiPriceList.INCLSELLERS**

Boolean

Value	Usage
<i>true</i> (default)	Include price lists with sellers assigned in the returned price list data.
<i>false</i>	Exclude price lists with sellers assigned from the returned price list data.

### **ccrz.ccApiPriceList.NAME**

String that specifies the name of a particular price list to query.

### **ccrz.ccApiPriceList.PLISERVSPPLIT**

Integer that specifies the maximum number of price list items that can pass through `ccrz.ccService.transform` at once.



**Tip:** The default value is 200. You can fine-tune performance of your price data queries by modifying this value.

### **ccrz.ccApiPriceList.PRGRID**

String that specifies the ID of a [price group](#) record to filter price lists by.



**Note:** If you don't include this key, the method falls back to `ccrz.cc_CallContext.priceGroupId`.



**ccrz.ccApiPriceList.PRODUCTIDS**

Set<String> of specific product IDs. The `ccrz.ccApiPriceList.PRICELISTITEMS` return data key includes price list items assigned to these products.

```
new Set<String>{'Product_ID_1', 'Product_ID_2'}
```

**ccrz.ccApiPriceList.PRODUCTSKUS**

Set<String> of specific product SKU values. The `ccrz.ccApiPriceList.PRICELISTITEMS` return data key includes price list items assigned to these products.

```
new Set<String>{'Product_SKU_1', 'Product_SKU_2'}
```

**ccrz.ccApiPriceList.SELLERS**

Set<String> of specific seller IDs whose assigned price lists you want to return:

```
new Set<String>{'Seller_ID_1', 'Seller_ID_2'}
```

**ccrz.ccApiPriceList.STOREFRONT**

String that specifies the name of a storefront for filtering the query.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiPriceList.SUBPRODTERMS**

Set<String> of specific subscription term IDs. The `ccrz.ccApiPriceList.PRICELISTITEMS` return data key includes price list items associated with these subscription terms.

```
new Set<String>{'SubProdTerm_ID_1', 'SubProdTerm_ID_2'}
```

**Important:** Specify only one key from among `ccrz.ccApiPriceList.IDS`, `ccrz.ccApiPriceList.EFFACCTID`, or `ccrz.ccApiPriceList.PRGRID`.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

**ccrz.ccApiPriceList.PRICELISTS**

Map<String, Map<String, Object>> keyed by the IDs of the returned price lists.


 **Tip:** To specify which fields the API returns for each price list record, [request a specific data size](#) for the `ccrz.ccApiPriceList.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiPriceList.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields are available with different data sizes, see the [ccrz\\_\\_E\\_PriceList\\_\\_c](#) reference.

#### **ccrz.ccApiPriceList.PRICELISTITEMS**

`Map<String, Map<String, Object>>` keyed by the IDs of the price list items that belong to the price lists in `ccrz.ccApiPriceList.PRICELISTS`. This key is returned only when you include `ccrz.ccApiPriceList.PRODUCTSKUS`, `ccrz.ccApiPriceList.PRODUCTIDS`, or `ccrz.ccApiPriceList.SUBPRODTERMS` in the input map. This method chains a call to the [ccrz.ccApiPriceList.fetchPriceListItems](#) method to return the price list items data.

 **Tip:** To specify which fields the API returns for each price list item, [request a specific data size](#) for the `ccrz.ccApiPriceList.ENTITYNAME_PLI` key in your input data. You can request unique sizes for price list data and price list item data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiPriceList.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
    },
    ccrz.ccApiPriceList.ENTITYNAME_PLI => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

You can also exclude the price list items from the return data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiPriceList.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
        ccrz.ccApi.SZ_ASSC => false
    }
}
```

#### **Example: Example**

For a specified price list ID and a set of specific product SKUs, fetch the price list data and its price list items.

```
Map<String, Object> priceListsToFetch = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPriceList.IDS => new Set<String>{'Price_List_ID'},
    ccrz.ccApiPriceList.PRODUCTSKUS => new Set<String>{'Product_SKU_1',
    'Product_SKU_2'},
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiPriceList.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
        },
        ccrz.ccApiPriceList.ENTITYNAME_PLI => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
        }
    }
}
```

```

    }
  }
};

Map<String, Object> priceListsFetched = ccrz.ccApiPriceList.fetch(priceListsToFetch);

System.debug('priceListsFetched KEYS=' +
JSON.serializePretty(priceListsFetched.keySet()));
Boolean success = (Boolean)priceListsFetched.get(ccrz.ccApi.SUCCESS);

if(success) {
  Map<String, Map<String, Object>> priceLists = (Map<String, Map<String,
Object>>)priceListsFetched.get(ccrz.ccApiPriceList.PRICELISTS);
  System.debug('pricelists=' + JSON.serializePretty(priceLists));

  Map<String, Map<String, Object>> priceListItems = (Map<String, Map<String,
Object>>)priceListsFetched.get(ccrz.ccApiPriceList.PRICELISTITEMS);
  System.debug('pricelistitems=' + JSON.serializePretty(priceListItems));
} else {
  List<ccrz.cc_bean_Message> messages =
(List<ccrz.cc_bean_Message>)priceListsFetched.get(ccrz.ccApi.MESSAGES);
  System.debug('messages=' + messages);
}
}

```

**SEE ALSO:**

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccApiPriceList.fetchEntitled

Executes a SOQL or SOSL query of price list records, and returns a list of price list IDs that the current account is entitled to. You can filter by account group or price group entitlements.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetchEntitled(Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicPLEntLists`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#). To return the price list and price list item records that correspond to the entitled price list IDs, include `ccrz.ccApi.SZ_REFETCH => true` in this key's map.

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiPriceList.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

### `ccrz.ccApiPriceList.ALLWMIX`

Boolean

Value	Usage
<code>true</code>	Allow mixed pricing queries when you filter by price group. The method tries to match the specified price group when returning price list records. If a price group match isn't available, the method falls back to the specified account's account group.
<code>false</code> (default)	Don't allow mixed pricing queries when you filter by price group. The method only tries to match the specified price group when returning price list records.

If this key isn't specified in the input data, the method falls back to the value of the `ccrz__E_AccountSettings__c.ccrz__AllowMixedPricingCarts__c` field.

### `ccrz.ccApiPriceList.CURRCODE`


String that specifies an ISO 4217 currency code, such as `USD` or `JPY`.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userCurrency`.


**ccrz.ccApiPriceList.EFFACCTID**

String that specifies an [account ID](#) record to filter entitled price lists by. The method returns only the price list records that are assigned to the specified account's account group.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.currAccountId`.


**ccrz.ccApiPriceList.EFFDATE**

Date to filter price lists by. The method returns only the price list records where the date is included in the Start Date and End Date range.

 **Note:** If the input data doesn't include this key, the method falls back to the current date.


**ccrz.ccApiPriceList.PRGRID**

String that specifies a [price group ID](#) record to filter price lists by.

 **Note:** If you don't include this key, the method falls back to `ccrz.cc_CallContext.priceGroupId`.

**ccrz.ccApiPriceList.STOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>


**ccrz.ccApiPriceList.PLIDS**

Map<String, Set<String>> keyed by:

- `ccrz.ccApiPriceList.PGPLIDS` for price lists entitled by price group
- `ccrz.ccApiPriceList.ACCTGRPLIDS` for price lists entitled by account group

**ccrz.ccApiPriceList.PRICELISTS**


Map<String, Map<String, Object>> keyed by the IDs of the entitled price lists.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

**ccrz.ccApiPriceList.PRICELISTITEMS**

Map<String, Map<String, Object>> keyed by the IDs of the price list items that belong to the price lists in ccrz.ccApiPriceList.PRICELISTS.

 **Note:** This key is returned only when the input map includes ccrz.ccApi.SZ\_REFETCH => true.

 **Note:** For more information about the ccrz.ccApiPriceList.PRICELISTS and ccrz.ccApiPriceList.PRICELISTITEMS keys, see [ccrz.ccApiPriceList.fetch](#).

**Example: Example**

Fetch price list IDs for a specified account ID.

```
Map<String, Object> entitledPriceListsToFetch = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPriceList.EFFACCTID => 'Account_ID'
};

Map<String, Object> entitledPriceListsFetched =
ccrz.ccApiPriceList.fetchEntitled(entitledPriceListsToFetch);

System.debug('entitledPriceListsFetched KEYS=' +
JSON.serializePretty(entitledPriceListsFetched.keySet()));
Boolean success = (Boolean)entitledPriceListsFetched.get(ccrz.ccApi.SUCCESS);

if(success) {
    Map<String, Object> priceListIDs = (Map<String,
Object>)entitledPriceListsFetched.get(ccrz.ccApiPriceList.PLIDS);
    System.debug('plids=' + JSON.serializePretty(priceLists));
} else {
    List<ccrz.cc_bean_Message> messages =
(List<ccrz.cc_bean_Message>)entitledPriceListsFetched.get(ccrz.ccApi.MESSAGES);
    System.debug('messages=' + messages);
}
```


**ccrz.ccApiPriceList.fetchPriceListItems**

Executes a SOQL query of price list item records, and returns a list of price list item IDs.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetchPriceListItems (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccServicePL`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiPriceList.CURRCODE`

String that specifies an ISO 4217 currency code, such as *USD* or *JPY*.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userCurrency`.

### `ccrz.ccApiPriceList.EFFACCTID`

String that specifies the ID of an [account](#) record to filter price list items by. The method returns only the price list items for the price lists that are assigned to the specified account's account group.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.currAccountId`.

### `ccrz.ccApiPriceList.EFFDATEPLI`

Date to filter price list items by. The method returns only the price list item records where the date is included in the Start Date and End Date range.

 **Note:** If the input data doesn't include this key, the method falls back to the current date.

### `ccrz.ccApiPriceList.IDS`

`Set<String>` of specific price list IDs whose items you want to return:

```
new Set<String>{ 'Price_List_ID_1', 'Price_List_ID_2' }
```


### `ccrz.ccApiPriceList.PLI_IDS`

`Set<String>` of specific price list item IDs to query:

```
new Set<String>{ 'Price_List_Item_ID_1', 'Price_List_Item_ID_2' }
```

**ccrz.ccApiPriceList.PLISERVSPPLIT**

Integer that specifies the maximum number of price list items that can pass through `ccrz.ccService.transform` at once.

 **Tip:** The default value is 200. You can fine-tune performance of your price data queries by modifying this value.

**ccrz.ccApiPriceList.PRODUCTIDS**

Set<String> of specific IDs for the product records whose price list items you want to query.

```
new Set<String>{ 'Product_ID_1', 'Product_ID_2' }
```

**ccrz.ccApiPriceList.PRODUCTSKUS**

Set<String> of specific SKU values for the product records whose price list items you want to query.

```
new Set<String>{ 'Product_SKU_1', 'Product_SKU_2' }
```

**ccrz.ccApiPriceList.NAME**

String that specifies the name of a price list whose items you want to return.

**ccrz.ccApiPriceList.STOREFRONT**


String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiPriceList.SUBPRODTERMS**

Set<String> of specific subscription term IDs whose price list items you want to query.

```
new Set<String>{ 'SubProdTerm_ID_1', 'SubProdTerm_ID_2' }
```

 **Important:** Specify only one key from among `ccrz.ccApiPriceList.IDS` or `ccrz.ccApiPriceList.EFFACCTID`.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**


Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiPriceList.PRICELISTITEMS**

Map<String, Map<String, Object>> keyed by the IDs of the returned price list items.



 **Tip:** To specify which fields the API returns for each price list item, [request a specific data size](#) for the `ccrz.ccApiPriceList.ENTITYNAME_PLI` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiPriceList.ENTITYNAME_PLI => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

For information about which fields are available with different data sizes, see the [ccrz\\_\\_E\\_PriceListItem\\_\\_c](#) reference.

### Example: Example

For a set of specific product SKUs, fetch the price list items assigned to those products.

```
Map<String, Object> priceListItemsToFetch = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPriceList.PRODUCTSKUS => new Set<String>{'Product_SKU_1',
    'Product_SKU_2'},
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiPriceList.ENTITYNAME_PLI => new Map<String, Object> {
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
        }
    }
};

Map<String, Object> priceListItemsFetched =
ccrz.ccApiPriceList.fetchPriceListItems(priceListItemsToFetch);

System.debug('priceListItemsFetched KEYS=' +
JSON.serializePretty(priceListItemsFetched.keySet()));
Boolean success = (Boolean)priceListItemsFetched.get(ccrz.ccApi.SUCCESS);

if(success) {
    Map<String, Map<String, Object>> priceListItems = (Map<String, Map<String,
Object>>)priceListItemsFetched.get(ccrz.ccApiPriceList.PRICELISTITEMS);
    System.debug('pricelistitems=' + JSON.serializePretty(priceListItems));
} else {
    List<ccrz.cc_bean_Message> messages =
(List<ccrz.cc_bean_Message>)priceListItemsFetched.get(ccrz.ccApi.MESSAGES);
    System.debug('messages=' + messages);
}
```

## ccrz.ccApiPrivateCache

Query, write, or update cache records for data that must stay private to your org, such as configuration data. For data that's safe for any storefront user to see—such as page labels or the category tree—use methods of the `ccrz.ccApiPublicCache` class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Cache Implementation

Methods of the `ccrz.ccApiPrivateCache` class use the following information from records of the `ccrz__E_PrivateCache__c` object:

### Cache Name

Specifies the type of data that the record caches, such as `ccConfig`. A Cache Name is required for all method calls. Because multiple private cache records can cache the same type of data, multiple private cache records can have the same Cache Name value.

### Build Group

Groups multiple private cache records together as part of the same build operation. For example, sometimes building the configuration cache from a batch job requires multiple batches to fully rebuild the configuration cache. Such a batch job creates multiple private cache records with the same Cache Name and Build Group. By associating multiple private cache records to a single build operation, you can more easily manage all the records related to that build.

### Storefront

Specifies a storefront that the cached data applies to. Sometimes, cached data can belong to all storefronts.

### Page


Specifies a page that the cached data applies to. Sometimes, cached data can belong to all pages.

### Locale

Specifies a locale that the cached data applies to. Cached data can apply to:

- A specific language and region, such as `en_US`
- Any locale with the same language component, such as `en`, where both `en_US` and `en_GB` locales apply
- Any locale

To write new private cache records, chain the `ccrz.ccApiPrivateCache.startWrite`, `ccrz.ccApiPrivateCache.appendTo`, and `ccrz.ccApiPrivateCache.close` methods.

 **Note:** The following example is simplified to demonstrate the general flow of operations. An actual implementation is more complex and evaluates all storefronts, pages, and locales.

```
Map<String, Object> cacheContext = ccrz.ccApiPrivateCache.startWrite(new Map<String, Object>{

    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPrivateCache.CACHE_NAME => 'ccConfig',
    ccrz.ccApiPrivateCache.LOCALE => 'en_US',
    ccrz.ccApiPrivateCache.STORE_NAME => 'Global',
    ccrz.ccApiPrivateCache.PAGENAME => 'All'

});
```

```
ccrz.ccApiPrivateCache.appendTo(new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPrivateCache.CONTEXT => cacheContext,
    ccrz.ccApiPrivateCache.CONTENT => 'Some content that is constructed elsewhere but is
just a string here'
});

ccrz.ccApiPrivateCache.close(new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPrivateCache.CONTEXT => cacheContext
});
```

 **Warning:** When writing or fetching cached data, limit the size of cached data to less than 2 MB.

## Methods

### IN THIS SECTION:

#### [fetch](#)

Executes a SOQL query of cached data by storefront, page, or locale, and returns the matching cache content as a String.

#### [startWrite](#)

Creates and inserts a new cache record, which is initially disabled. This method doesn't create any actual content for the cache, but returns a *cache context*, which is required for subsequent `ccrz.ccApiPrivateCache.appendTo` and `ccrz.ccApiPrivateCache.close` method calls.

#### [appendTo](#)

Adds content to a private cache record created with the `ccrz.ccApiPrivateCache.startWrite` method. Call this method as often as necessary to populate the cache. This method returns a *cache context*, which you pass to subsequent `ccrz.ccApiPrivateCache.appendTo` and `ccrz.ccApiPrivateCache.close` method calls.

#### [close](#)

Stops writing any more content to the private cache record created with the `ccrz.ccApiPrivateCache.startWrite` and `ccrz.ccApiPrivateCache.appendTo` methods. If you passed `ccrz.ccApiPrivateCache.ENABLE_ON_CLOSE => true` in the input key for `ccrz.ccApiPrivateCache.startWrite`, the cache is enabled after `ccrz.ccApiPrivateCache.close` completes.

#### [disableOld](#)

Disables cache records that have the same cache name, storefront, page, and locale as another cache record with a more recent Effective Start date and time. This method changes the value of the Enabled field on the affected private cache records. Make sure that you specify a cache name, storefront, page name, and locale in the method's input map.

#### [toggleGroup](#)

Changes the value of the Enabled field for all private cache records that belong to the same build group.

### SEE ALSO:

[ccrz.ccApiPublicCache](#)


## ccrz.ccApiPrivateCache.fetch

Executes a SOQL query of cached data by storefront, page, or locale, and returns the matching cache content as a String.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

### Service Layer Classes


This method doesn't support service layer extensions or overrides.

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### **ccrz.ccApi.API\_VERSION**


The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### **ccrz.ccApiPrivateCache.CACHE\_NAME**

String that specifies a Cache Name for the private cache records to query, such as `ccConfig`.

```
ccrz.ccApiPrivateCache.CACHE_NAME => 'ccConfig'
```


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPrivateCache.NoCacheNameException`.

### Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApiPrivateCache.LOCALE**

String that specifies a locale to filter the queried cached data by.

 **Note:** If cached data isn't available for the specified locale, the method falls back to query other locales with the same language component. For example, maybe you specify `en_US` but cached data isn't available for that locale. The method then queries for cached data from any locale that starts with `en`, such as `en_GB` or `en_AU`.

**ccrz.ccApiPrivateCache.PAGENAME**

String that specifies the name of a page whose cached data you want to query. Use the page keys defined in [CC Admin global settings](#), such as `hp` for Home page and `p1p` for Product List page.

```
ccrz.ccApiPrivateCache.PAGENAME => 'hp'
```

To query cached data from all pages, specify `All`. If you don't include this key in your input map, the method uses `All` by default.

```
ccrz.ccApiPrivateCache.PAGENAME => 'All'
```

**ccrz.ccApiPrivateCache.STORE\_NAME**

String that specifies a storefront whose cached data you want to query.

```
ccrz.ccApiPrivateCache.STORE_NAME => 'DefaultStore'
```

To query cached data from all storefronts, specify `Global`. If you don't include this key in your input map, the method uses `Global` by default.

```
ccrz.ccApiPrivateCache.STORE_NAME => 'Global'
```

## Fallback Behavior

Depending on which values you include in the input map, this method incrementally falls back to a less specific query, in the following order of priority.

 **Note:** The Sequence Priority field on a private cache record specifies the relative priority of that record in query results.

Priority	ccrz.ccApiPrivateCache.STORE_NAME Value	ccrz.ccApiPrivate.PAGENAME Value	ccrz.ccApiPrivateCache.LOCALE Value
1	The exact value specified	The exact value specified	The exact value specified
2	The exact value specified	The exact value specified	Any locale with the same language component
3	The exact value specified	The exact value specified	None
4	The exact value specified	<code>All</code>	The exact value specified
5	The exact value specified	<code>All</code>	Any locale with the same language component
6	The exact value specified	<code>All</code>	None
7	<code>Global</code>	The exact value specified	The exact value specified
8	<code>Global</code>	The exact value specified	Any locale with the same language component

Priority	<code>ccrz.ccApiPrivateCache.STORE_NAME</code> Value	<code>ccrz.ccApiPrivate.PAGENAME</code> Value	<code>ccrz.ccApiPrivateCache.LOCALE</code> Value
9	<i>Global</i>	The exact value specified	None
10	<i>Global</i>	<i>All</i>	The exact value specified
11	<i>Global</i>	<i>All</i>	Any locale with the same language component
12	<i>Global</i>	<i>All</i>	None

## Outputs


Map<String, Object> that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

### `ccrz.ccApiPrivateCache.CONTENT`

String of the cached data that matches the query.

## `ccrz.ccApiPrivateCache.startWrite`


Creates and inserts a new cache record, which is initially disabled. This method doesn't create any actual content for the cache, but returns a *cache context*, which is required for subsequent `ccrz.ccApiPrivateCache.appendTo` and `ccrz.ccApiPrivateCache.close` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> startWrite (Map<String, Object>)
```

## Service Layer Classes


This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`


The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiPrivateCache.CACHE_NAME`

String that specifies the name of cached data to insert a new cache record for, such as `ccConfig`.

```
ccrz.ccApiPrivateCache.CACHE_NAME => 'ccConfig'
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPrivateCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiPrivateCache.BUILD_GROUP`

String that specifies the build group for the cache, which groups all cache records built at approximately the same time for simpler maintenance.

### `ccrz.ccApiPrivateCache.BUILD_TIMESTAMP`

Datetime for the build group, which identifies when the cache record was built. For all cache records that belong to the same build group, use the same timestamp.

### `ccrz.ccApiPrivateCache.EFFECTIVE_END`

Datetime when the cache is no longer valid. If you don't include this key in your input map, the method assigns a datetime 50 years from the current date and time.

**ccrz.ccApiPrivateCache.EFFECTIVE\_START**

Datetime when the cache record becomes valid. If you don't include this key in your input map, the method uses the constant *NOW*, which specifies the current date and time.

**ccrz.ccApiPrivateCache.ENABLE\_ON\_CLOSE**

Boolean

Value	Usage
<i>true</i>	Enable the cache record after the <code>ccrz.ccApiPrivateCache.close</code> method is complete.
<i>false</i> (default)	The cache record remains disabled after it's created.

**ccrz.ccApiPrivateCache.LOCALE**

String that specifies a locale to insert a new private cache record for. If you don't include this key in your input map, the new cache record applies to all locales.

**ccrz.ccApiPrivateCache.PAGENAME**

String that specifies the name of a page to insert a new private cache record for. If you don't include this key in your input map, the new cache record applies to all pages.

**ccrz.ccApiPrivateCache.STORE\_NAME**

String that specifies a storefront to insert a new private cache record for. If you don't include this key in your input map, the new cache record applies to all storefronts.

## Outputs


Map<*String*, *Object*> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The call completed.
<i>false</i>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <i>false</i>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiPrivateCache.CONTEXT**

`ccrz.ccApiPrivateCache.CacheContext` object that you pass to `ccrz.ccApiPrivateCache.appendTo` and `ccrz.ccApiPrivateCache.close` method calls for this cache record.




## ccrz.ccApiPrivateCache.appendTo

Adds content to a private cache record created with the `ccrz.ccApiPrivateCache.startWrite` method. Call this method as often as necessary to populate the cache. This method returns a *cache context*, which you pass to subsequent `ccrz.ccApiPrivateCache.appendTo` and `ccrz.ccApiPrivateCache.close` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> appendTo (Map<String, Object>)
```

### Service Layer Classes


This method doesn't support service layer extensions or overrides.

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`


The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### `ccrz.ccApiPrivateCache.CACHE_NAME`

String that specifies the name of cached data to add, such as `ccConfig`.

```
ccrz.ccApiPrivateCache.CACHE_NAME => 'ccConfig'
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPrivateCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApiPrivateCache.CONTEXT**

`ccrz.ccApiPrivateCache.CacheContext` object returned from `ccrz.ccApiPrivateCache.startWrite` or previous `ccrz.ccApiPrivateCache.appendTo` method calls.

### **ccrz.ccApiPrivateCache.CONTENT**

String that specifies the content to add to the cache record.

## Outputs


Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### **ccrz.ccApiPrivateCache.CONTEXT**

`ccrz.ccApiPrivateCache.CacheContext` object that you pass to `ccrz.ccApiPrivateCache.appendTo` and `ccrz.ccApiPrivateCache.close` method calls for this cache record.

## ccrz.ccApiPrivateCache.close


Stops writing any more content to the private cache record created with the `ccrz.ccApiPrivateCache.startWrite` and `ccrz.ccApiPrivateCache.appendTo` methods. If you passed `ccrz.ccApiPrivateCache.ENABLE_ON_CLOSE => true` in the input key for `ccrz.ccApiPrivateCache.startWrite`, the cache is enabled after `ccrz.ccApiPrivateCache.close` completes.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> close(Map<String, Object>)
```

## Service Layer Classes


This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`


The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiPrivateCache.CACHE_NAME`

String that specifies the name of cached data to close, such as `ccConfig`.

```
ccrz.ccApiPrivateCache.CACHE_NAME => 'ccConfig'
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPrivateCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiPrivateCache.CONTEXT`

`ccrz.ccApiPrivateCache.CacheContext` object returned from [ccrz.ccApiPrivateCache.appendTo](#) method calls.

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint.</a>


## ccrz.ccApiPrivateCache.disableOld

Disables cache records that have the same cache name, storefront, page, and locale as another cache record with a more recent Effective Start date and time. This method changes the value of the Enabled field on the affected private cache records. Make sure that you specify a cache name, storefront, page name, and locale in the method's input map.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> disableOld (Map<String, Object>)
```

### Service Layer Classes

This method doesn't support service layer extensions or overrides.

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**


The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

**ccrz.ccApiPrivateCache.CACHE\_NAME**

String that specifies the name of cached data to disable, such as `ccConfig`.

```
ccrz.ccApiPrivateCache.CACHE_NAME => 'ccConfig'
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPrivateCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApiPrivateCache.LOCALE**

String that specifies a locale to disable old cache entries for. If locale doesn't apply to the cache entries you want to disable, pass an empty string:

```
ccrz.ccApiPrivateCache.LOCALE => ''
```

**ccrz.ccApiPrivateCache.PAGENAME**

String that specifies a page to disable old cache entries for.

**ccrz.ccApiPrivateCache.STORE\_NAME**

String that specifies a storefront to disable old cache entries for.

## Outputs

None


## ccrz.ccApiPrivateCache.toggleGroup

Changes the value of the Enabled field for all private cache records that belong to the same build group.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> toggleGroup(Map<String, Object>)
```

## Service Layer Classes


This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`


The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiPrivateCache.CACHE_NAME`

String that specifies a Cache Name value for the private cache records you want to update, such as `ccConfig`.

```
ccrz.ccApiPrivateCache.CACHE_NAME => 'ccConfig'
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPrivateCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:


### `ccrz.ccApiPrivateCache.BUILD_GROUP`

String that specifies a Build Group value for the private cache records you want to update. This method updates all private cache records with the same `ccrz.ccApiPrivateCache.CACHE_NAME` and `ccrz.ccApiPrivateCache.BUILD_GROUP` values.

### `ccrz.ccApiPrivateCache.SET_ENABLED`

Boolean

Value	Usage
<code>true</code>	Enable the Enabled field on each affected private cache record.
<code>false</code>	Disable the Enabled field on each affected private cache record.

 **Note:** If you don't include this key, by default this method changes the current value of the Enabled field to the opposite value. For example, if Enabled is true, this method changes Enabled to false.

## Outputs

None

## ccrz.ccApiProduct

Query product records, price entitled products, and return other entitled product details, including records related to each product, such as composite products and product media.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

#### [fetch](#)

Executes a SOQL query of product records, and returns a list of products that match your query. This method can also return multiple lists of related object records, such as composite products and product media.

#### [find](#)

Returns and prices entitled products that match a search string or that belong to a set of specified category IDs.

#### [findFilters](#)

Returns filters and spec values for entitled products that match a search string or that belong to a set of specified category IDs.

#### [getUrlFor](#)

Returns a map where each entry associates a product's ID with its friendly URL.

#### [revise](#)

Updates the value of only the Product Index Status field on existing product records. This method doesn't support updating other fields on product records.

#### [search](#)

Executes a SOSL query of product records, and returns the IDs of products that match the search query.


## ccrz.ccApiProduct.fetch

Executes a SOQL query of product records, and returns a list of products that match your query. This method can also return multiple lists of related object records, such as composite products and product media.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceProduct`

### Logic Service Provider


`ccrz.ccLogicProductPricing`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

### `ccrz.ccApiProduct.PRODUCTIDLIST`

`Set<String>` of specific product IDs to query:


```
new Set<String>{'Product_ID_1', 'Product_ID_2'}
```

### `ccrz.ccApiProduct.PRODUCTSKULIST`

`Set<String>` of specific product SKUs to query:

```
new Set<String>{'Product_SKU_1', 'Product_SKU_2'}
```



 **Note:** If you specify both keys, `ccrz.ccApiProduct.PRODUCTIDLIST` takes precedence.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

Map<[String](#), [Object](#)> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### `ccrz.ccApiProduct.DATEFILTER`

Boolean

Value	Usage
<code>true</code> (default)	Evaluate Start Date and End Date values when returning product data.
<code>false</code>	Ignore Start Date and End Date values when returning product data.

### `ccrz.ccApiProduct.INCLUDE_ATTRIBUTE_PRICING`

Boolean

Value	Usage
<code>true</code>	Evaluate attribute data when <code>ccrz.ccApiProduct.PARAM_INCLUDE_PRICING</code> is also <code>true</code> . B2B Commerce for Visualforce calls the <code>ccrz.ccLogicProductPricing</code> logic service class, which: <ul style="list-style-type: none"> <li>• Queries for <code>ccrz__E_Attribute__c</code> records that are related to each product</li> <li>• Returns the attribute pricing JSON applied to the product's entitled price list item</li> </ul>
<code>false</code> (default)	Don't return any attribute pricing.

### `ccrz.ccApiProduct.MEDIAFILTER`

Set<[String](#)> of specific product media types that you want to return in the related `ccrz__E_ProductMedia__c` records.

### `ccrz.ccApiProduct.PARAM_BY_ASC`

Boolean

Value	Usage
<code>true</code>	Sort returned products by ID in ascending order. For example, sort from 100 to 900. If you pass <code>ccrz.ccApiProduct.PARAM_BY_SEQ =&gt; true</code> , products sort by Sequence value in ascending order.
<code>false</code> (default)	Sort returned products by ID in descending order.

### `ccrz.ccApiProduct.PARAM_BY_NULLS_LAST`

Boolean

Value	Usage
<i>true</i>	Place all products with a null Sequence value at <a href="#">the end of the returned product data</a> .
<i>false</i> (default)	Place all products with a null Sequence value at the beginning of the returned product data.

**ccrz.ccApiProduct.PARAM\_BY\_SEQ**

Boolean

Value	Usage
<i>true</i>	Sort returned products by Sequence value in descending order. For example, sort from 900 to 100.
<i>false</i> (default)	Sort returned products by ID in descending order.

**ccrz.ccApiProduct.PARAM\_INCLUDE\_BLOCKED**

Boolean

Value	Usage
<i>true</i>	Return products with any status.
<i>false</i> (default)	Return products with a status of only Released or Not Orderable.

**ccrz.ccApiProduct.PARAM\_INCLUDE\_PRICING**

Boolean

Value	Usage
<i>true</i> (default)	Include <code>ccrz.ccApiProduct.PRODUCTPRICINGDATA</code> in the product return data. B2B Commerce for Visualforce calls the <code>ccrz.ccLogicProductPricing</code> logic service class to evaluate each product's entitlements when determining pricing.
<i>false</i>	Don't return any pricing data.

**ccrz.ccApiProduct.PARAM\_INCLUDE\_SELLERS**

Boolean

Value	Usage
<i>true</i> (default)	Return a list of all sellers for each product. Make sure that <code>ccrz.ccApiProduct.PARAM_INCLUDE_PRICING</code> is also <i>true</i> .
<i>false</i>	Don't return any seller information.

**ccrz.ccApiProduct.PARSE\_ATTRIBUTE\_PRICING**

Boolean

Value	Usage
<code>true</code> (default)	Parse the attribute pricing JSON returned when <code>ccrz.ccApiProduct.INCLUDE_ATTRIBUTE_PRICING</code> is <code>true</code> .
<code>false</code>	Skip querying attribute records and return only an unparsed attribute pricing field from each product's entitled price list item.

**ccrz.ccApiProduct.PRODUCTLOCALE**

String that specifies one of your storefront's supported locales. The method returns internationalized product values in the `ccrz.ccApiProduct.PRODUCTLIST` return key. The service layer translates the values from `ccrz__E_ProductItemI18N__c` records and handles fallback languages, if necessary.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccApiProduct.PRODUCTSTOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiProduct.TYPES**

Set<String> of specific product types to query for.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean


Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

**ccrz.ccApiAttribute.IDS**

List<String> of entitled attribute records to pass to the `ccrz.ccApiAttribute.fetch` method for returning attributes related to the returned products. This data is available only when you include `ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true` in the input map.

**ccrz.ccApiProduct.PRODUCTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_Product__c` that matches the query.

 **Tip:** To specify which fields the API returns for each product record, [request a specific data size](#) for the `ccrz.ccApiProduct.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```


For information about which fields and related lists (such as product media) are available with different data sizes, see the [ccrz\\_\\_E\\_Product\\_\\_c](#) reference.

#### **ccrz.ccApiProduct.PRODUCTPRICINGDATA**

`Map<String, Object>` of the pricing data returned when you include `ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true` in the input map. This map contains seller, subscription term, attribute, and standard pricing information for the returned products.

#### **ccrz.ccApiSpec.SPECLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a [ccrz\\_\\_E\\_Spec\\_\\_c](#) record associated with product spec values, which are included in the `productSpecs` field of `ccrz.ccApiProduct.PRODUCTLIST`. For more information about fetching spec data, see [ccrz.ccApiSpec.fetch](#).

 **Tip:** To specify which fields the API returns for each spec record, [request a specific data size](#) for the `ccrz.ccApiSpec.ENTITYNAME` key in your input data. You can request unique sizes for product data and spec data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    },
    ccrz.ccApiSpec.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

By default, the `ccrz.ccApiSpec.SPECLIST` is returned when the input map requests a large data size for `ccrz.ccApiProduct.ENTITYNAME`. You can optionally exclude the spec records from the return data.

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L,
        ccrz.ccApi.SZ_ASSC => false
    }
}
```

#### **ccrz.ccApiSubProdTerm.SPTLIST**

`List<Map<String, Object>>` of subscription product terms related to the returned products.

#### **ccrz.ccApiSubProdTerm.SPTBYPRODID**

`Map<String, Object>` of subscription product terms per product.

### **Example: Example**

Query a pair of product IDs, and capture the first product's name in a variable.

```
Map<String, Object> productFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
```

```

ccrz.ccApiProduct.PRODUCTIDLIST => new Set<String>{'Product_ID_1', 'Product_ID_2'},

ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
    }
}
};

try {
    Map<String, Object> productFetchData = ccrz.ccApiProduct.fetch(productFetchQuery);

    if (productFetchData.get(ccrz.ccApiProduct.PRODUCTLIST) != null) {
        List<Map<String, Object>> myProductDataList = (List<Map<String, Object>>)
productFetchData.get(ccrz.ccApiProduct.PRODUCTLIST);
        String productName = (String) myProductDataList[0].get('sfdcName');
    }
} catch (Exception e) {
    // Error handling...
}

```

## Platform Cache

By default, this method checks for product data previously queried and stored in Platform Cache. This design helps improve performance on your storefront when displaying your product data.

The underlying `ccrz.ccServiceProduct.fetch` method that supports the `ccrz.ccApiProduct.fetch` method checks the Platform Cache for product data.

1. Before the `ccrz.ccApiProduct.fetch` method executes the query for product data, the underlying `ccrz.ccServiceProduct.fetch` data service method calls the `ccrz.ccService.prepareCacheKey` method.
2. If `ccrz.ccService.prepareCacheKey` finds a match in Platform Cache for the queried data, it passes the matching data to the `ccrz.ccService.prepReturn` method, which sends the data to the `ccrz.ccApiProduct.fetch` query.

If no matching data is found in the Platform Cache, the `ccrz.ccApiProduct.fetch` query executes and returns data from your org's database. In this case, the `ccrz.ccService.prepareCacheData` method is called to store the product data in Platform Cache.

 **Note:** Storing product data in Platform Cache also stores related data, such as product media and product specs, all of which consume space allocated in the cache.

To specify how long B2B Commerce for Visualforce keeps product data cached, [change the product data time to live \(TTL\) values in your storefront configuration settings](#).

### SEE ALSO:

- [ccrz.ccLogicProductPricing](#)
- [Salesforce Help Set Up Attribute-Based Products](#)
- [Salesforce Help Set Up Product Data for B2B Commerce for Visualforce](#)
- [Salesforce Help Host a Seller Marketplace on Your Storefront](#)
- [Salesforce Help Offer Subscriptions for Products on Your Storefront](#)
- [Improve Storefront Performance with Platform Cache](#)


## ccrz.ccApiProduct.find

Returns and prices entitled products that match a search string or that belong to a set of specified category IDs.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> find(Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider


`ccrz.ccLogicProductFind`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

#### `ccrz.ccApiProduct.CATEGORY_IDS`

`Set<String>` of specific category IDs whose assigned products you want to return and price.

#### `ccrz.ccService.SEARCHSTRING`

String to use for searching product records, and whose results you want to return and price.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApiProduct.CURRENT\_INDEX**

Map<String, Object> that specifies the product index information for the product to use as a starting point in the `ccrz.ccApiProduct.find` call.

```
Map<String, Object> findProductsStartingFrom = new Map<String, Object> {
    primary => 'String of product's sfdcName',
    secondary => CC Product Index record ID
}
```

You can use the `ccrz.ccApiProduct.CURRENT_INDEX`, `ccrz.ccApiProduct.PRODUCT_LIMIT`, and `ccrz.ccService.ORDERBY` keys to chain `ccrz.ccApiProduct.find` calls together. For details, see [this method's example](#) on page 524.

### **ccrz.ccApiProduct.INCLUDE\_COUNT**

Boolean


Value	Usage
<code>true</code>	Return the total count of products, keyed under <code>ccrz.ccApiProduct.COUNT</code> .
<code>false</code> (default)	Don't return product count.

### **ccrz.ccApiProduct.PRODUCT\_FILTERS**

List<Map<String, Object>>, where each Map<String, Object> specifies spec values to filter products by.

### **ccrz.ccApiProduct.PRODUCT\_LIMIT**

Integer that specifies the maximum number of product IDs to return.

 **Note:** If you specify a lower limit than the actual number of results, the return data includes the index information for the next indexed product. For example, if you specify a limit of five, the return data includes the index information for the sixth product that matches the find criteria.

### **ccrz.ccApiProduct.PRODUCTLOCALE**

String that specifies one of your [storefront's supported locales](#). The method returns internationalized product values in the `ccrz.ccApiProduct.PRODUCTLIST` return key. The service layer translates the values from `ccrz__E_ProductItemI18N__c` records and handles fallback languages, if necessary.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

### **ccrz.ccApiProduct.PRODUCTSTOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### **ccrz.ccService.ORDERBY**

String that specifies the field to order results by. Possible values include:

- `ccrz.ccApiProductIndex.BY_NAME`
- `ccrz.ccApiProductIndex.BY_PRICE`
- `ccrz.ccApiProductIndex.BY_START_DATE`
- `ccrz.ccApiProductIndex.BY_RELEVANCE`

- `ccrz.ccApiProductIndex.BY_SEQUENCE` (default)

**ccrz.ccService.SORTDESC**

Boolean

Value	Usage
<code>true</code>	Sort returned products by the field that the <code>ccrz.ccService.ORDERBY</code> key specifies, in descending order.
<code>false</code> (default)	Sort returned products in ascending order.

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, consider rolling back the API transaction to a previous savepoint.</p>

**ccrz.ccApiProduct.PRODUCTLIST**

`List<Object>` where each object represents a `ccrz__E_Product__c` record that matches the find criteria.

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**

`Map<String, Object>` of pricing data, keyed by product ID. The map contains seller, subscription term, attribute, and standard pricing information for the returned products.

 **Example: Example**

Specify a limit of 10 results returned for a search string, and sort the results by product name. Then, execute a second `ccrz.ccApiProduct.find` call for another set of 10 results, starting with the 11th result from the first call.

```
Map<String, Object> firstGroupOfProductsToFind = new Map<String, Object> {
    ccrz.ccApiProduct.PRODUCT_LIMIT => 10,
    ccrz.ccService.SEARCHSTRING => 'Some string to search products for',
    ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_NAME
};

Map<String, Object> = firstGroupOfProductsFound =
ccrz.ccApiProduct.find(firstGroupOfProductsToFind);

Map<String, Object> secondGroupOfProductsToFind = new Map<String, Object> {
```



```

ccrz.ccApiProduct.CURRENT_INDEX =>
firstGroupOfProductsFound.get(ccrz.ccApiProduct.CURRENT_INDEX),
ccrz.ccApiProduct.PRODUCT_LIMIT => 10,
ccrz.ccService.SEARCHSTRING = 'Some string to search products for',
ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_NAME
};

```

SEE ALSO:

[ccrz.ccApiProductIndex](#)


## ccrz.ccApiProduct.findFilters

Returns filters and spec values for entitled products that match a search string or that belong to a set of specified category IDs.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> findFilters(Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider


[ccrz.ccLogicProductFindFilters](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

**ccrz.ccApiProduct.CATEGORY\_IDS**

`Set<String>` of specific category IDs whose assigned products you want to return product filters for.

**ccrz.ccService.SEARCHSTRING**

String to use for searching product index records for returning product filters. The `ccrz.ccApiProductIndex.search` method executes the search with a SOSL query.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApiProduct.PRODUCT\_FILTERS**

`List<Map<String, Object>>`, where each `Map<String, Object>` specifies a spec ID and product spec value to filter products by:

```
[ {
  "specValues" : [ {
    "value" : "Blue"
  } ],
  "sfid" : "a1h6A00000045s5QAA"
} ]
```

**ccrz.ccApiProduct.PRODUCTLOCALE**

String that specifies one of your [storefront's supported locales](#). The method returns internationalized product spec values in the `ccrz.ccApiProduct.PRODUCT_FILTERS` return key.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

## Outputs

`Map<String, Object>` that can include the following keys:


**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### `ccrz.ccApiProduct.PRODUCT_FILTERS`

Map<String, Map<String, Object>> of aggregated product filters, keyed by spec value. When serialized to JSON, this map of filters uses the following format:

```
[
  {
    "specValues": [
      {
        "count": 100,
        "value": "SpecValA"
      }
    ],
    "filterType": "Checkbox",
    "sequence": 200,
    "sfid": "alh6A000001ERMvQAO",
    "name": "SpecA0"
  }
]
```


### `ccrz.ccApiProduct.getUrlFor`

Returns a map where each entry associates a product's ID with its friendly URL.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> getUrlFor (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


```
ccrz.ccLogicProductUrlFor
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include one of the following keys. If none of these keys are included, the method returns a `ccrz.ccApi.MissingInputException`.

### **ccrz.ccApiProduct.PRODUCTIDLIST**

`List<String>` of Salesforce IDs for specific products whose friendly URLs you want to return.

### **ccrz.ccApiProduct.PRODUCTLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_Product__c` whose friendly URL you want to return.

### **ccrz.ccApiProduct.PRODUCTSKULIST**

`List<String>` of specific product SKUs whose friendly URLs you want to return.

 **Note:** This method checks for these keys in the following order of precedence:

1. `ccrz.ccApiProduct.PRODUCTLIST`
2. `ccrz.ccApiProduct.PRODUCTSKULIST`
3. `ccrz.ccApiProduct.PRODUCTIDLIST`

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApiProduct.PRODUCTLOCALE**

String that specifies one of your [storefront's supported locales](#). The method returns internationalized friendly URL values in the `ccrz.ccApiProduct.PRODUCTURLMAP` return key.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

### **ccrz.ccApiProduct.PRODUCTLOCALES**

`Set<String>` of your [storefront's supported locales](#). The method returns internationalized friendly URL values for the locales.

### **ccrz.ccApiProduct.STOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs


Map<String, Object> that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiProduct.PRODUCTURLMAP`

Map<String, Object> for a single locale, where each key is a product ID and each value is the product's friendly URL.

## Example

Query a pair of product IDs and return their friendly URL values.

```
try{
    // Gather set of products
    Map<String, Object> productReturnData = ccrz.ccApiProduct.fetch(new Map<String, Object>{

        ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
        ccrz.ccApiProduct.PRODUCTIDLIST => new List<String>{'Product_ID_1',
'Product_ID_2'},
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object>{
            ccrz.ccAPI.SZ_DATA => ccrz.ccAPI.SZ_S,
            ccrz.ccAPI.SZ_ASSC => false,
            ccrz.ccAPI.SZ_REL => new List<String>{'Product_Item_I18Ns__r'}
        }
    });

    // Pass products to getUrlFor
    Map<String, Object> urlReturnData = ccrz.ccApiProduct.getUrlFor(new Map<String, Object>{

        ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
        ccrz.ccApiProduct.PRODUCTLIST => productReturnData.get(ccrz.ccApiProduct.PRODUCTLIST)
    });

    // Grab the URL map from the return data
```

```

    Map<String, Object> productUrlMap = (Map<String,
Object>)urlReturnData.get(ccrz.ccApiProduct.PRODUCTURLMAP);
} catch(Exception e){
    // Error handling
}

```

SEE ALSO:

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)


## ccrz.ccApiProduct.revise

Updates the value of only the Product Index Status field on existing product records. This method doesn't support updating other fields on product records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

 **Warning:** This method doesn't allow a guest user to update records. If you're creating subscriber code that requires a guest user to update records, you have the following options:

- If you're using B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11), invoke the [ccrz.ccLogic.updateObjects utility method](#).
- Execute the DML operation in your custom Apex class using the [without sharing accessor](#).

### Signature

```
global static Map<String, Object> revise(Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider

ccrz.ccServiceProduct

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiProduct.PRODUCTLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the `sfid` of a `ccrz__E_Product__c` to update and the `productIndexStatus` field value you want to set.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiProduct.fetch` for returning the updated products.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.


### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiProduct.PRODUCTLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` represents an updated `ccrz__E_Product__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

 **Example: Example**

Change the `productIndexStatus` field value for a specified product record.

```
Map<String, Object> productToUpdate = new Map<String, Object>{
    'sfid' => 'Product_ID',
    'productIndexStatus' => 'Current'
};

Map<String, Object> inputData = (new Map<String, Object>{
    ccrz.ccApiProduct.PRODUCTLIST => new List<Map<String, Object>>{productToUpdate},
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApi.SIZING => new Map<String, Object> {
        ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
            ccrz.ccApi.SZ_REFETCH => true
        }
    }
});

try {
    Map<String, Object> outputData = ccrz.ccApiProduct.revise(inputData);
} catch (Exception e) {
    // Error handling...
}
```


## ccrz.ccApiProduct.search

Executes a SOSL query of product records, and returns the IDs of products that match the search query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.



## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

```
ccrz.ccServiceProduct
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccService.SEARCHSTRING`

String to use for searching product records.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiProduct.PRODUCTLOCALE`

String that specifies one of your [storefront's supported locales](#). The method returns internationalized product records in the return data.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

### `ccrz.ccApiProduct.PRODUCTSTOREFRONT`

String that specifies the name of a storefront for filtering the query.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

`Map<String, Object>` that can include the following keys:


### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiProduct.PRODUCTSEARCHMAP**

Map<String, String> that relates each returned product's ID to its SKU.

**ccrz.ccService.SEARCHRESULTS**

Set<String> of product IDs that match the search string.

 **Example: Example**

Search products for the string *coffee grinder*, and then call `ccrz.ccApiProduct.fetch` on the IDs that the search returned.

```
Map<String, Object> productSearchInput = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiProduct.PRODUCTSTOREFRONT => 'DefaultStore',
    ccrz.ccService.SEARCHSTRING => 'coffee grinder'
};

try {
    Map<String, Object> productSearchResults =
ccrz.ccApiProduct.search(productSearchInput);
    if (productSearchResults.get(ccrz.ccService.SEARCHRESULTS) != null) {
        Set<String> searchIdSet = (Set<String>)
productSearchResults.get(ccrz.ccService.SEARCHRESULTS);

        // Call search with the set?
        Map<String, Object> fetchproductSearchInput = new Map<String, Object>{
            ccrz.ccApiProduct.PRODUCTIDLIST => searchIdSet,
            ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION
        };

        try {
            Map<String, Object> productSearchResults =
ccrz.ccApiProduct.fetch(productSearchInput);
            if (productSearchResults.get(ccrz.ccApiProduct.PRODUCTLIST) != null) {
                // The cast to List<Map<String, Object>> is necessary...
                List<Map<String, Object>> outputProductList = (List<Map<String, Object>>)
productSearchResults.get(ccrz.ccApiProduct.PRODUCTLIST);

                // The cast, again, is necessary...
                String productName = (String) outputProductList[0].get('sfdcName');
            }
        } catch (Exception e) {
            // Error handling...
        }
    }
} catch (Exception e) {
```

```
// Error handling...
}
```

## ccrz.ccApiProductIndex

Query, create, update, or remove product index records, which B2B Commerce for Visualforce uses for quickly populating content on the Product List page version 2.0.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

#### IN THIS SECTION:

##### [create](#)

Creates product index records from price list items that you specify. This method creates product index records for each locale your storefront supports and for each category that a product belongs to.

##### [fetch](#)

Executes a SOQL query of product index records and returns a list of records that match your query.

##### [fetchEntitled](#)

Executes a SOQL or SOSL query of product index records and returns a list of product IDs that the current user is entitled to. You can further filter which entitled product IDs the method returns by a search term, specific category assignments, product spec values, and specific price lists.

##### [remove](#)

Delete product index records. B2B Commerce for Visualforce uses this method in product index batch jobs.

##### [revise](#)

Updates the status of the Active field on existing product index records. B2B Commerce for Visualforce uses this method in product index batch jobs.

##### [search](#)

Executes a SOSL search of product index records and returns a list of records that match your search.


## ccrz.ccApiProductIndex.create

Creates product index records from price list items that you specify. This method creates product index records for each locale your storefront supports and for each category that a product belongs to.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> create (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

[ccrz.ccLogicPCreate](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include one of the following keys. If none of these keys are included, the method returns a `ccrz.ccApi.MissingInputException`.

#### **ccrz.ccApiProductIndex.PRICE\_LIST\_ITEM\_IDS**

`Set<String>` of Salesforce IDs of [price list item](#) records that you want to create product index records for.

```
new Set<String>{'Price_List_Item_ID_1', 'Price_List_Item_ID_2'}
```

**ccrz.ccApiProductIndex.PRICE\_LIST\_ITEMS**

Map<String, Object> of price list item records, keyed by ID, that you want to create product index records for.

**ccrz.ccApiProductIndex.PRODUCT\_IDS**

Set<String> of specific product IDs that you want to create product index records for.

```
new Set<String>{'Product_ID_1', 'Product_ID_2'}
```

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiProductIndex.fetch` for returning the created product index records.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
  ccrz.ccApiProductIndex.ENTITYNAME => new Map<String, Object>{
    ccrz.ccApi.SZ_REFETCH => TRUE
  }
}
```

**ccrz.ccApiProductIndex.MASTER\_PRICE\_LIST**

String that specifies the ID of the storefront's [master price list](#), if applicable.

Set<String> of locales that you want to create product index records for.

```
new Set<String>{'en_US', 'ja_JP'}
```



**Note:** If the input data doesn't specify this key, this method queries all locales.

## Outputs

Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean


Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiProductIndex.IDS**

Set<String> of the created product index records.

**ccrz.ccApiProductIndex.PRODUCT\_INDICES**

List<Map<String, Object>>, where each Map<String, Object> represents a created [product index record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

 **Example: Example**

Creates product index records for three specific price list items.

```
Map<String, Object> priceListItemsToCreate = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiProductIndex.PRICE_LIST_ITEM_IDS => new Set<String>{'a176A000000t612',
    'a176A000000t613', 'a176A000000t614'}
};

try {
    Map<String, Object> createProductIndexRecords =
    ccrz.ccApiProductIndex.create(priceListItemsToCreate);
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

[ccrz.ccProductIndexBuilder](#)

[Salesforce Help Maintain Product Index Records](#)


## ccrz.ccApiProductIndex.fetch

Executes a SOQL query of product index records and returns a list of records that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

```
ccrz.ccServiceProductIndex
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### `ccrz.ccApiProductIndex.ACTIVE`

Boolean

Value	Usage
<code>true</code>	Return only product index records where the Active field is enabled.
<code>false</code> (default)	Return product index records that match your other filter criteria, regardless of the Active field value.

### `ccrz.ccApiProductIndex.CATEGORY_IDS`

`Set<String>` of specific category IDs whose product index records you want to query:

```
new Set<String>{ 'Category_ID_1', 'Category_ID_2' }
```

### `ccrz.ccApiProductIndex.CURRENT_INDEX`

`Map<String, Object>` that specifies a starting point for the method's query by indicating the last product index record within the previous set of sorted results. You can pass this map to subsequent `ccrz.ccApiProductIndex.fetch`, `ccrz.ccApiProductIndex.search`, or `ccrz.ccApiProductIndex.fetchEntitled` method calls for querying the next set of results. This map specifies the following subkeys:

Key	Usage
primary	The value that the <code>ccrz.ccService.ORDERBY</code> input key used for sorting product index records.
secondary	The ID of the referenced product index record.

For example:

```
{
  "primary": "sort value, such as index price or product name",
  "secondary": "Product_Index_ID"
}
```

#### **ccrz.ccApiProductIndex.DATE\_FILTER**

Boolean

Value	Usage
<i>true</i>	Return only product index records where the Start Date and End Date range includes the current date.
<i>false</i> (default)	Return product index records that match your other filter criteria, regardless of date range.

#### **ccrz.ccApiProductIndex.IDS**

Set<String> of specific product index IDs to query:

```
new Set<String>{ 'Product_Index_ID_1', 'Product_Index_ID_2' }
```

#### **ccrz.ccApiProductIndex.LOCALE**

String that specifies a locale to filter product index records by.

#### **ccrz.ccApiProductIndex.PRICE\_LISTS**

Set<String> of specific price list IDs whose associated product index records you want to query:

```
new Set<String>{ 'Price_List_ID_1', 'Price_List_ID_2' }
```

#### **ccrz.ccApiProductIndex.PRODUCT\_FILTERS**

Map<String, Object> that specifies spec names and product IDs that have values defined for the specs. The method returns product index records that match the specified spec names and product IDs.

```
ccrz.ccApiProductIndex.PRODUCT_FILTERS => new Map<String, Object>{
  'color' => 'Product_ID_1', 'Product_ID_2'
  'width' => 'Product_ID_3', 'Product_ID_4'
}
```

#### **ccrz.ccApiProductIndex.STATUS\_FILTER**

Boolean

Value	Usage
<i>true</i>	Return only product index records where the referenced product has a Status field value of Released or Not Orderable.



Value	Usage
<code>false</code> (default)	Return product index records that match your other filter criteria, regardless of product status.

**ccrz.ccApiProductIndex.STOREFRONT**

String that specifies a storefront to filter product index records by.


 **Note:** Your version of B2B Commerce for Visualforce determines how the method filters by storefront:

Version	Behavior
B2B Commerce for Visualforce Summer '19 (managed package version 4.11 and API version 10) or later	Queries the Storefront field directly on a <a href="#">product index record</a> .
B2B Commerce for Visualforce Spring '19 (managed package version 4.10 and API version 9) or earlier	Queries the Storefront field on a product that a product index record references.

**ccrz.ccApiProductIndex.TYPE**

String that specifies a value for the Index Type field to filter product index records by. By default, the only supported value is the key `ccrz.ccApiProductIndex.INDEX_TYPE_SEARCH`, which corresponds to the Index Type value `UOVBUkNI`, the Base64-encoded value for the string `Search`.


```
ccrz.ccApiProductIndex.TYPE => ccrz.ccApiProductIndex.INDEX_TYPE_SEARCH
```



 **Note:** B2B Commerce for Visualforce requires this format for identifying products assigned to multiple categories and ensuring that products appear only once in a search. When a product index record specifies `UOVBUkNI` for its Index Type, that product index is the record returned in search results.

**ccrz.ccService.ORDERBY**

String that specifies a key, which indicates how you want to sort product index records. For example:

```
ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_NAME
```

Value	Usage
<code>ccrz.ccApiProductIndex.BY_NAME</code>	Sort by the localized name of the product that the product index record references.
<code>ccrz.ccApiProductIndex.BY_PRICE</code>	Sort by the Index Price field on the product index record.
<code>ccrz.ccApiProductIndex.BY_RELEVANCE</code>	Sort by occurrences of the search string in the product's name, short description, and SKU.   <b>Note:</b> <ul style="list-style-type: none"> <li>Relevance sorting is available only for product index records that match a search string.</li> <li>When this key is specified, this method calls into the <a href="#">ccrz.ccLogicPIFetchEntitled.rankSearchResultsAndProcess</a> method for executing the relevance sorting.</li> </ul>

Value	Usage
<code>ccrz.ccApiProductIndex.BY_SEQUENCE</code>	Sort by the Sequence field on the <a href="#">product category record</a> that the product index record references.   <b>Note:</b> <ul style="list-style-type: none"> <li>Sequence sorting is available only for product index records that match specified category IDs.</li> <li>B2B Commerce for Visualforce Summer '19 (managed package version 4.11, API version 10) or later flattens the data model required for referencing this Sequence value. The product index record includes a Product Category Sequence field, whose value is copied from the corresponding product category record's Sequence field.</li> </ul>
<code>ccrz.ccApiProductIndex.BY_START_DATE</code>	Sort by the value of the Start Date field on the product that the product index record references.   <b>Note:</b> B2B Commerce for Visualforce Summer '19 (managed package version 4.11, API version 10) or later flattens the data model required for referencing this Start Date value. The product index record includes a Product Start Date field, whose value is copied from the corresponding product record's Start Date field.

**ccrz.ccService.SORTDESC**

Boolean

Value	Usage
<code>true</code>	Sort product index records by the field that <code>ccrz.ccService.ORDERBY</code> specifies in descending order.
<code>false</code> (default)	Sort product index records by the field that <code>ccrz.ccService.ORDERBY</code> specifies in ascending order.

**Outputs**

Map<String, Object> that can include the following keys:


**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**


Boolean

Value	Usage
<code>true</code>	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### ccrz.ccApiProductIndex.PRODUCT\_INDICES

List<Map<String, Object>>, where each Map<String, Object> represents a [product index record](#).

 **Tip:** To specify which fields the API returns for each product index record, [request a specific data size](#) for the `ccrz.ccApiCart.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProductIndex.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields are available with different data sizes, see the [ccrz\\_\\_E\\_ProductIndex\\_\\_c](#) reference.

### Example: Examples

Query product index records for a specific category and locale.

```
Map<String, Object> fetchInputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiProductIndex.CATEGORY_IDS => new List<String>{'a176A000000t612'},
    ccrz.ccApiProductIndex.LOCALE => 'en_US'
};

try {
    Map<String, Object> fetchReturnData = ccrz.ccApiProductIndex.fetch(fetchInputData);
} catch (Exception e) {
    // Error handling...
}
```

Query product index records for a specific category and locale. Order the returned product index records by the Indexed Price field, and start the query from a previously queried product index.

```
Map<String, Object> fetchInputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiProductIndex.CATEGORY_IDS => new List<String>{'a176A000000t612'},
    ccrz.ccApiProductIndex.LOCALE => 'en_US',
    ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_PRICE,
    ccrz.ccApiProductIndex.CURRENT_INDEX => new Map<String, Object>{
        'primary' => '000000000008627.00',
        'secondary' => 'a176A000000t611'
    }
};

try {
```

```

    Map<String, Object> fetchReturnData = ccrz.ccApiProductIndex.fetch(fetchInputData);
} catch (Exception e) {
    // Error handling...
}

```


## ccrz.ccApiProductIndex.fetchEntitled

Executes a SOQL or SOSL query of product index records and returns a list of product IDs that the current user is entitled to. You can further filter which entitled product IDs the method returns by a search term, specific category assignments, product spec values, and specific price lists.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetchEntitled(Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider


[ccrz.ccLogicPIFetchEntitled](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

**ccrz.ccApiProductIndex.CATEGORY\_IDS**

Set<String> of specific category IDs passed to the `ccrz.ccApiProductIndex.fetch` method, which executes a SOQL query of product index records that reference the specified category IDs.

**ccrz.ccService.SEARCHSTRING**

String passed to the `ccrz.ccApiProductIndex.search` method for executing a SOSL query of product index records.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApiProductIndex.CURRENT\_INDEX**

Map<String, Object> that specifies a starting point for the method's query by indicating the last product index record within the previous set of sorted results. You can pass this map to subsequent `ccrz.ccApiProductIndex.fetch`, `ccrz.ccApiProductIndex.search`, or `ccrz.ccApiProductIndex.fetchEntitled` method calls for querying the next set of results. This map specifies the following subkeys:

Key	Usage
primary	The value that the <code>ccrz.ccService.ORDERBY</code> input key used for sorting product index records.
secondary	The ID of the referenced product index record.

For example:

```
{
  "primary": "sort value, such as index price or product name",
  "secondary": "Product_Index_ID"
}
```

**ccrz.ccApiProductIndex.INCLUDE\_COUNT**

Boolean

Value	Usage
true	Include the <code>ccrz.ccApiProductIndex.COUNT</code> key in the return data, which indicates the total number of product index records that match the query.
false (default)	Don't include the <code>ccrz.ccApiProductIndex.COUNT</code> key in the return data.

**ccrz.ccApiProductIndex.INDEX\_LIMIT**

Integer that specifies a maximum number of product index records to query, which affects how many product IDs are returned.

**ccrz.ccApiProductIndex.PRICE\_LISTS**

Set<String> of price list IDs, which filters the product index query for only those records that reference the specified price lists. If you exclude this key from the input map, the method invokes the `ccrz.ccApiPriceList.fetchEntitled` method and filters the query by all the user's entitled price lists.

**ccrz.ccApiProductIndex.PRODUCT\_IDS\_ONLY**


Boolean

Value	Usage
<i>true</i>	Return only <code>ccrz.ccApiProductIndex.PRODUCT_IDS</code> , and skip calculating <code>ccrz.ccApiProductIndex.CURRENT_INDEX</code> .
<i>false</i> (default)	Calculate <code>ccrz.ccApiProductIndex.CURRENT_INDEX</code> for the return data.

**ccrz.ccApiProductIndex.PRODUCT\_SPEC\_FILTERS**

List<Map<String, Object>> or List<Object> of specs and spec values to pass to the `ccrz.ccApiSpec.PRODUCT_SPEC_INDICES` key for the [ccrz.ccApiSpec.fetchProductSpecIndices](#) method. Include each spec's ID and the values that you want to filter by, such as:

```
[ {
  "specValues" : [ {
    "value" : "Blue"
  } ],
  "sfid" : "alh6A00000045s5QAA"
} ]
```

 **Note:** If you specify multiple values for the same spec, the method applies an OR condition for evaluating all values.


For a spec where Filter Type is Slider, you can specify a `filterMin` and `filterMax` range for filtering:



```
[ {
  "sfid" : "alh6A00000045s5QAA",
  "filterMin" : "1.0",
  "filterMax" : "10.0"
} ]
```

**ccrz.ccService.ORDERBY**

String that specifies a key, which indicates how you want to sort product index records. For example:

```
ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_NAME
```

Value	Usage
<code>ccrz.ccApiProductIndex.BY_NAME</code>	Sort by the localized name of the product that the product index record references.
<code>ccrz.ccApiProductIndex.BY_PRICE</code>	Sort by the Index Price field on the product index record.
<code>ccrz.ccApiProductIndex.BY_RELEVANCE</code>	Sort by occurrences of the search string in the product's name, short description, and SKU.   <b>Note:</b> <ul style="list-style-type: none"> <li>Relevance sorting is available only for product index records that match a search string.</li> <li>When this key is specified, this method calls into the <a href="#">ccrz.ccLogicPIFetchEntitled.rankSearchResultsAndProcess</a> method for executing the relevance sorting.</li> </ul>

Value	Usage
<code>ccrz.ccApiProductIndex.BY_SEQUENCE</code>	Sort by the Sequence field on the <a href="#">product category record</a> that the product index record references.   <b>Note:</b> <ul style="list-style-type: none"> <li>Sequence sorting is available only for product index records that match specified category IDs.</li> <li>B2B Commerce for Visualforce Summer '19 (managed package version 4.11, API version 10) or later flattens the data model required for referencing this Sequence value. The product index record includes a Product Category Sequence field, whose value is copied from the corresponding product category record's Sequence field.</li> </ul>
<code>ccrz.ccApiProductIndex.BY_START_DATE</code>	Sort by the value of the Start Date field on the product that the product index record references.   <b>Note:</b> B2B Commerce for Visualforce Summer '19 (managed package version 4.11, API version 10) or later flattens the data model required for referencing this Start Date value. The product index record includes a Product Start Date field, whose value is copied from the corresponding product record's Start Date field.

## Outputs


Map<[String](#), [Object](#)> that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.


### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiProductIndex.COUNT`

Integer that indicates the total number of product index records that match the query.

 **Note:** This key is returned only when the input data specifies `ccrz.ccApiProductIndex.INCLUDE_COUNT => true`.

**ccrz.ccApiProductIndex.CURRENT\_INDEX**

Map<String, Object> that specifies the last product index record within the current set of sorted results. You can pass this map to subsequent `ccrz.ccApiProductIndex.fetch`, `ccrz.ccApiProductIndex.search`, or `ccrz.ccApiProductIndex.fetchEntitled` method calls for querying the next set of results. This map specifies the following subkeys:

Key	Usage
primary	The value that the <code>ccrz.ccService.ORDERBY</code> input key used for sorting product index records.
secondary	The ID of the referenced product index record.

For example:

```
{
  "primary": "sort value, such as index price or product name",
  "secondary": "Product_Index_ID"
}
```

**ccrz.ccApiProductIndex.HAS\_MORE**

Boolean

Value	Usage
true	The method returned more results than the limit that the <code>ccrz.ccApiProductIndex.INDEX_LIMIT</code> input key specified.
false	The method returned fewer results than the limit that the <code>ccrz.ccApiProductIndex.INDEX_LIMIT</code> input key specified.

**ccrz.ccApiProductIndex.PRODUCT\_CATEGORY\_SEQUENCE**

Map<String, Decimal> that relates each entitled product ID to its corresponding product category sequence value. This key is returned only when the input data includes:

```
ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_SEQUENCE
```

**ccrz.ccApiProductIndex.PRODUCT\_FILTERS**

Map<String, List<String>> that relates a spec ID with the entitled product IDs that have spec values for that spec. The method returns this key when the input map includes `ccrz.ccApiProductIndex.PRODUCT_SPEC_FILTERS`.

**ccrz.ccApiProductIndex.PRODUCT\_IDS**

List<String> of entitled product IDs from the queried product index records. The number of product IDs included in this list depends on whether the input data specified a `ccrz.ccApiProductIndex.INDEX_LIMIT` value.

**ccrz.ccApiProductIndex.PRODUCT\_SEARCH\_RANKINGS**

Map<String, Integer> that relates each entitled product ID to its determined relevance rank. For example:

```
{
  "an18digitProductID" : 100,
  "an18digitProductID" : 200,
  "an18digitProductID" : 250,
}
```



```
"an18digitProductID" : 50
}
```

The method returns this key only when you include the `ccrz.ccService.SEARCHSTRING` key in the input map.


## ccrz.ccApiProductIndex.remove

Delete product index records. B2B Commerce for Visualforce uses this method in product index batch jobs.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> remove (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

`ccrz.ccLogicPIRemove`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`


The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### `ccrz.ccApiProductIndex.IDS`

`Set<String>` of Salesforce IDs of product index records to delete.

```
new Set<String>{ 'Product_Index_ID_1', 'Product_Index_ID_2' }
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .


## `ccrz.ccApiProductIndex.revise`

Updates the status of the Active field on existing product index records. B2B Commerce for Visualforce uses this method in product index batch jobs.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

 **Warning:** This method doesn't allow a guest user to update records. If you're creating subscriber code that requires a guest user to update records, you have the following options:

- If you're using B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11), invoke the `ccrz.ccLogic.updateObjects` utility method.
- Execute the DML operation in your custom Apex class using the `without sharing` accessor.

## Signature

```
global static Map<String, Object> revise (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider


```
ccrz.ccServiceProductIndex
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiProductIndex.PRODUCT_INDICES`

`List<Map<String, Object>>`, where each `Map<String, Object>` contains a product index ID and the Active field value to set.

```
Map<String, Object> productIndexToUpdate = new Map<String, Object>{
    'sfid' => 'a176A000000t6l2',
    'active' => FALSE
};
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiProductIndex.fetch` for returning the updated product index records.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiProductIndex.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiProductIndex.PRODUCT\_INDICES**

List<Map<String, Object>>, where each Map<String, Object> represents an updated [product index record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

 **Example: Example**

Update the value of the Active field on two product index records to `FALSE`.

```
Map<String, Object> productIndex1 = new Map<String, Object>{
    'sfid' => 'a176A000000t6l2',
    'active' => FALSE
};

Map<String, Object> productIndex2 = new Map<String, Object>{
    'sfid' => 'a176A000s00t6l2',
    'active' => FALSE
};

Map<String, Object> productIndicesToRevise = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiProductIndex.PRODUCT_INDICES => new List<Map<String, Object>>{productIndex1,
    productIndex2}
};

try {
    Map<String, Object> reviseReturnData =
    ccrz.ccApiProductIndex.revise(productIndicesToRevise);
} catch (Exception e) {
    // Error handling...
}
```


**ccrz.ccApiProductIndex.search**

Executes a SOSL search of product index records and returns a list of records that match your search.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> search (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceProductIndex`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiProductIndex.ACTIVE`

Boolean

Value	Usage
<code>true</code>	Return only product index records where the Active field is enabled.
<code>false</code> (default)	Return product index records that match your other filter criteria, regardless of the Active field value.

**ccrz.ccApiProductIndex.CURRENT\_INDEX**

Map<String, Object> that specifies a starting point for the method's query by indicating the last product index record within the previous set of sorted results. You can pass this map to subsequent `ccrz.ccApiProductIndex.fetch`, `ccrz.ccApiProductIndex.search`, or `ccrz.ccApiProductIndex.fetchEntitled` method calls for querying the next set of results. This map specifies the following subkeys:

Key	Usage
primary	The value that the <code>ccrz.ccService.ORDERBY</code> input key used for sorting product index records.
secondary	The ID of the referenced product index record.

For example:

```
{
  "primary": "sort value, such as index price or product name",
  "secondary": "Product_Index_ID"
}
```

**ccrz.ccApiProductIndex.LOCALE**

String that specifies a locale to filter product index records by.

**ccrz.ccApiProductIndex.STOREFRONT**

String that specifies a storefront to filter product index records by.



**Note:** Your version of B2B Commerce for Visualforce determines how the method filters by storefront:

Version	Behavior
B2B Commerce for Visualforce Summer '19 (managed package version 4.11 and API version 10) or later	Queries the Storefront field directly on a <a href="#">product index record</a> .
B2B Commerce for Visualforce Spring '19 (managed package version 4.10 and API version 9) or earlier	Queries the Storefront field on a product that a product index record references.

**ccrz.ccApiProductIndex.PRICE\_LISTS**

Set<String> of specific price list IDs whose associated product index records you want to search:

```
new Set<String>{ 'Price_List_ID_1', 'Price_List_ID_2' }
```

**ccrz.ccApiProductIndex.PRODUCT\_FILTERS**

Map<String, Object> that specifies spec names and product IDs that have values defined for the specs. The method returns product index records that match the specified spec names and product IDs.

```
ccrz.ccApiProductIndex.PRODUCT_FILTERS => new Map<String, Object>{
  'color' => 'Product_ID_1', 'Product_ID_2'
  'width' => 'Product_ID_3', 'Product_ID_4'
}
```

**ccrz.ccApiProductIndex.TYPE**

String that specifies a value for the Index Type field to filter product index records by. By default, the only supported value is the key `ccrz.ccApiProductIndex.INDEX_TYPE_SEARCH`, which corresponds to the Index Type value `U0VBukNI`, the Base64-encoded value for the string `Search`.

```
ccrz.ccApiProductIndex.TYPE => ccrz.ccApiProductIndex.INDEX_TYPE_SEARCH
```



**Note:** B2B Commerce for Visualforce requires this format for identifying products assigned to multiple categories and ensuring that products appear only once in a search. When a product index record specifies `U0VBukNI` for its Index Type, that product index is the record returned in search results.

**ccrz.ccService.ORDERBY**

String that specifies a key, which indicates how you want to sort product index records. For example:

```
ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_NAME
```

Value	Usage
<code>ccrz.ccApiProductIndex.BY_NAME</code>	Sort by the localized name of the product that the product index record references.
<code>ccrz.ccApiProductIndex.BY_PRICE</code>	Sort by the Index Price field on the product index record.
<code>ccrz.ccApiProductIndex.BY_RELEVANCE</code>	Sort by occurrences of the search string in the product's name, short description, and SKU. <p> <b>Note:</b></p> <ul style="list-style-type: none"> <li>Relevance sorting is available only for product index records that match a search string.</li> <li>When this key is specified, this method calls into the <a href="#">ccrz.ccLogicPIFetchEntitled.rankSearchResultsAndProcess</a> method for executing the relevance sorting.</li> </ul>
<code>ccrz.ccApiProductIndex.BY_SEQUENCE</code>	Sort by the Sequence field on the <a href="#">product category record</a> that the product index record references. <p> <b>Note:</b></p> <ul style="list-style-type: none"> <li>Sequence sorting is available only for product index records that match specified category IDs.</li> <li>B2B Commerce for Visualforce Summer '19 (managed package version 4.11, API version 10) or later flattens the data model required for referencing this Sequence value. The product index record includes a Product Category Sequence field, whose value is copied from the corresponding product category record's Sequence field.</li> </ul>
<code>ccrz.ccApiProductIndex.BY_START_DATE</code>	Sort by the value of the Start Date field on the product that the product index record references. <p> <b>Note:</b> B2B Commerce for Visualforce Summer '19 (managed package version 4.11, API version 10) or later flattens the data model required for referencing this Start Date value. The product index record includes a Product Start Date</p>

Value	Usage
	field, whose value is copied from the corresponding product record's Start Date field.

**ccrz.ccService.SEARCHSTRING**

String to search for among product index records. The method adds a trailing wildcard to the string.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiProductIndex.PRODUCT\_INDICES**

List<Map<String, Object>>, where each Map<String, Object> represents a [product index record](#).

 **Example: Example**

Search product index records for the string `espresso`. Order the returned product index records by the Indexed Price field.

```
Map<String, Object> searchInputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccService.SEARCHSTRING => 'espresso',
    ccrz.ccApiProductIndex.TYPE => ccrz.ccApiProductIndex.INDEX_TYPE_SEARCH,
    ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_PRICE
};

try {
    Map<String, Object> searchReturnData =
ccrz.ccApiProductIndex.search(searchInputData);
} catch (Exception e) {
    // Error handling...
}
```



## ccrz.ccApiPromotion

Query promotion records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

IN THIS SECTION:

#### [fetch](#)

Executes a SOQL query of promotion records and returns a list of records that match your query. By default, this method filters promotion records by the values of the Enabled, Start Date, and End Date fields.


### ccrz.ccApiPromotion.fetch

Executes a SOQL query of promotion records and returns a list of records that match your query. By default, this method filters promotion records by the values of the Enabled, Start Date, and End Date fields.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

```
ccrz.ccServicePromotion
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#).



**Note:** Specific data size requests don't apply to this method. The same fields are always returned for each object, regardless of any specific data size request in your input map.

### `ccrz.ccApiPromotion.FILTERED`

Boolean

Value	Usage
<code>true</code>	For each promotion that matches your other search criteria, evaluate any <a href="#">account group filters</a> that exist. Return only promotions that are visible for the current user's account group (see the following table).
<code>false</code> (default)	Return all promotions that match your other query criteria.

When this key is `true`, return only promotions that are visible for the current user's account group, using the following criteria:

Promotion Excludes Current User's Account Group?	Promotion Excludes Any Other Account Group?	Promotion Includes Current User's Account Group?	Promotion Includes Any Other Account Group?	Promotion is Visible for Current User?
No	No	No	No	Yes
No	Yes	No	No	Yes
No	Yes or No	Yes	Yes or No	Yes

Promotion Excludes Current User's Account Group?	Promotion Excludes Any Other Account Group?	Promotion Includes Current User's Account Group?	Promotion Includes Any Other Account Group?	Promotion is Visible for Current User?
Yes	Yes or No	Yes or No	Yes or No	No
No	Yes or No	No	Yes	No

 **Note:**

- The `ccrz.ccCallContext` determines the current user.
- The `ccrz.ccLogicPromotionFilter` class defines this logic. To modify this default behavior, you can extend this class with subscriber code.

**ccrz.ccApiPromotion.PROMOTIONLOCALE**

String that specifies a locale to return localized promotion data for in the `ccrz.ccApiPromotion.PROMOTIONLIST` return key. The service layer translates the values from `ccrz__E_PromoI18n__c` records and handles fallback languages, if necessary.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccApiPromotion.PROMOTIONLOCATION**

Set<String> of specific [Page Location values](#) to filter promotion records by:

```
new Set<String>{'Landing Page', 'Product List Page'}
```

**ccrz.ccApiPromotion.PROMOTIONSTOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiPromotion.PROMOTIONTYPE**

Set<String> of specific [Location Type values](#) to filter promotion records by:

```
new Set<String>{'Splash', 'Right Nav'}
```

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .


**ccrz.ccApiPromotion.PROMOTIONLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [promotion record](#).

 **Note:** Each promotion record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.

**ccrz.ccApiProduct.PRODUCTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [product record](#).

 **Note:** The method returns this key if `ccrz.ccApiPromotion.PROMOTIONLIST` includes a promotion where `LinkType` is `Product`. The `ccrz.ccApiProduct.PRODUCTLIST` includes details for the targeted product. The method also returns other applicable data related to those products, such as subscription terms, specs, related products, and more. For more information about fetching product data, see [ccrz.ccApiProduct.fetch](#).

 **Example: Example**

Query for promotions on the `DefaultStore` storefront where `Location Type` is `Left Nav`.

```
Map<String, Object> inputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPromotion.PROMOTIONSTOREFRONT => 'DefaultStore',
    ccrz.ccApiPromotion.PROMOTIONTYPE => new Set<String>{'Left Nav'}
};
try {
    Map<String, Object> outputData = ccrz.ccApiPromotion.fetch(inputData);
    if (outputData.get(ccrz.ccApiPromotion.PROMOTIONLIST) != null) {
        // The cast to List<Map<String, Object>> is necessary...
        List<Map<String, Object>> promotionList = (List<Map<String, Object>>)
outputData.get(ccrz.ccApiPromotion.PROMOTIONLIST);

        for(Map<String, Object> promotion : promotionList){
            // The cast, again, is necessary...
            String locationType = (String)promotion.get('locationType');
        }
    }
} catch (Exception e) {
    // Error handling...
}
```

## ccrz.ccApiPublicCache

Query, write, or update cache records for data that's safe for any storefront user to see, such as page labels or the category tree. To query, write, or update cache records for data that must stay private to your org—such as configuration data—use methods of the `ccrz.ccApiPrivateCache` class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Cache Implementation

Methods of the `ccrz.ccApiPublicCache` class use the following information from [public cache records](#):

### Cache Name

Specifies the type of data that the record caches, such as `ccPageLabels` or `ccCategoryTree`. A Cache Name is required for all method calls. Because multiple public cache records can cache the same type of data, multiple public cache records can have the same Cache Name value.

### Build Group

Groups multiple public cache records together as part of the same build operation. For example, sometimes building the page label cache from a batch job requires multiple batches to fully rebuild the page label cache. Such a batch job creates multiple public cache records with the same Cache Name and Build Group. By associating multiple public cache records to a single build operation, you can more easily manage all the records related to that build.

### Storefront

Specifies a storefront that the cached data applies to. Sometimes, cached data can belong to all storefronts.

### Page


Specifies a page that the cached data applies to. Sometimes, cached data can belong to all pages.

### Locale

Specifies a locale that the cached data applies to. Cached data can apply to:

- A specific language and region, such as `en_US`
- Any locale with the same language component, such as `en`, where both `en_US` and `en_GB` locales apply
- Any locale

To write new public cache records, chain the `ccrz.ccApiPublicCache.startWrite`, `ccrz.ccApiPublicCache.appendTo`, and `ccrz.ccApiPublicCache.close` methods.

 **Note:** The following example is simplified to demonstrate the general flow of operations. An actual implementation is more complex and evaluates all storefronts, pages, and locales.

```
Map<String, Object> cacheContext = ccrz.ccApiPublicCache.startWrite(new Map<String, Object>{

    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPublicCache.CACHE_NAME => 'ccPageLabels',
    ccrz.ccApiPublicCache.LOCALE => 'en_US',
    ccrz.ccApiPublicCache.STORE_NAME => 'Global',
    ccrz.ccApiPublicCache.PAGENAME => 'All'
});

ccrz.ccApiPublicCache.appendTo(new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPublicCache.CONTEXT => cacheContext,
```

```

    ccrz.ccApiPublicCache.CONTENT => 'Some content is constructed elsewhere but is just a
    string here'
  });

ccrz.ccApiPublicCache.close(new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiPublicCache.CONTEXT => cacheContext
  });

```

 **Warning:** When writing or fetching cached data, limit the size of cached data to less than 2 MB.

## Methods

### IN THIS SECTION:

#### [fetch](#)

Executes a SOQL query of cached data by storefront, page, or locale, and returns the matching cache content as a String.

#### [startWrite](#)

Creates and inserts a new cache record, which is initially disabled. This method doesn't create any actual content for the cache, but returns a *cache context*, which is required for subsequent `ccrz.ccApiPublicCache.appendTo` and `ccrz.ccApiPublicCache.close` method calls.

#### [appendTo](#)

Adds content to a public cache record created with the `ccrz.ccApiPublicCache.startWrite` method. Call this method as often as necessary to populate the cache. This method returns a *cache context*, which you pass to subsequent `ccrz.ccApiPublicCache.appendTo` and `ccrz.ccApiPublicCache.close` method calls.

#### [close](#)

Stops writing any more content to the public cache record created with the `ccrz.ccApiPublicCache.startWrite` and `ccrz.ccApiPublicCache.appendTo` methods. If you passed `ccrz.ccApiPublicCache.ENABLE_ON_CLOSE => true` in the input key for `ccrz.ccApiPublicCache.startWrite`, the cache is enabled after `ccrz.ccApiPublicCache.close` completes.

#### [disableOld](#)

Disables cache records that have the same cache name, storefront, page, and locale as another cache record with a more recent Effective Start date and time. This method changes the value of the Enabled field on the affected public cache records. Make sure that you specify a cache name, storefront, page name, and locale in the method's input map.

#### [toggleGroup](#)

Changes the value of the Enabled field for all public cache records that belong to the same build group.

### SEE ALSO:

[ccrz.ccApiPrivateCache](#)


## ccrz.ccApiPublicCache.fetch

Executes a SOQL query of cached data by storefront, page, or locale, and returns the matching cache content as a String.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes


This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiPublicCache.CACHE_NAME`

String that specifies a Cache Name for the public cache records to query, such as:

- `ccCategoryTree` for cached page label data that `ccrz.ccCategoryCacheBuildJob` builds.
- `ccPageLabel` for cached page label data that `ccrz.ccPageLabelCacheBuildJob` builds.

```
ccrz.ccApiPublicCache.CACHE_NAME => 'ccPageLabel'
```


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPublicCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApiPublicCache.LOCALE**

String that specifies a locale to filter the queried cached data by.

 **Note:** If cached data isn't available for the specified locale, the method falls back to query other locales with the same language component. For example, maybe you specify `en_US` but cached data isn't available for that locale. The method then queries for cached data from any locale that starts with `en`, such as `en_GB` or `en_AU`.

**ccrz.ccApiPublicCache.PAGENAME**

String that specifies the name of a page whose cached data you want to query. Use the page keys defined in [CC Admin global settings](#), such as `hp` for Home page and `p1p` for Product List page.

```
ccrz.ccApiPublicCache.PAGENAME => 'hp'
```

To query cached data from all pages, specify `All`. If you don't include this key in your input map, the method uses `All` by default.

```
ccrz.ccApiPublicCache.PAGENAME => 'All'
```

**ccrz.ccApiPublicCache.STORE\_NAME**

String that specifies a storefront whose cached data you want to query.

```
ccrz.ccApiPublicCache.STORE_NAME => 'DefaultStore'
```

To query cached data from all storefronts, specify `Global`. If you don't include this key in your input map, the method uses `Global` by default.

```
ccrz.ccApiPublicCache.STORE_NAME => 'Global'
```

## Fallback Behavior

Depending on which values you include in the input map, this method incrementally falls back to a less specific query, in the following order of priority.

 **Note:** The Sequence Priority field on a [public cache record](#) specifies the relative priority of that record in query results.

Priority	ccrz.ccApiPublicCache.STORE_NAME Value	ccrz.ccApiPublicCache.PAGENAME Value	ccrz.ccApiPublicCache.LOCALE Value
1	The exact value specified	The exact value specified	The exact value specified
2	The exact value specified	The exact value specified	Any locale with the same language component
3	The exact value specified	The exact value specified	None
4	The exact value specified	<code>All</code>	The exact value specified
5	The exact value specified	<code>All</code>	Any locale with the same language component
6	The exact value specified	<code>All</code>	None
7	<code>Global</code>	The exact value specified	The exact value specified
8	<code>Global</code>	The exact value specified	Any locale with the same language component



Priority	ccrz.ccApiPublicCache.STORE_NAME Value	ccrz.ccApiPublicCache.PAGENAME Value	ccrz.ccApiPublicCache.LOCALE Value
9	<i>Global</i>	The exact value specified	None
10	<i>Global</i>	<i>All</i>	The exact value specified
11	<i>Global</i>	<i>All</i>	Any locale with the same language component
12	<i>Global</i>	<i>All</i>	None

## Outputs


Map<String, Object> that can include the following keys:

### ccrz.ccApi.API\_VERSION

Integer that indicates which API version was used for the query.

### ccrz.ccApi.SUCCESS

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

### ccrz.ccApiPublicCache.CONTENT

String of the cached data that matches the query.

## ccrz.ccApiPublicCache.startWrite


Creates and inserts a new cache record, which is initially disabled. This method doesn't create any actual content for the cache, but returns a *cache context*, which is required for subsequent `ccrz.ccApiPublicCache.appendTo` and `ccrz.ccApiPublicCache.close` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> startWrite (Map<String, Object>)
```

## Service Layer Classes


This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiPublicCache.CACHE_NAME`

String that specifies the name of cached data to insert a new cache record for, such as:

- `ccCategoryTree` for cached page label data that `ccrz.ccCategoryCacheBuildJob` builds.
- `ccPageLabel` for cached page label data that `ccrz.ccPageLabelCacheBuildJob` builds.

```
ccrz.ccApiPublicCache.CACHE_NAME => 'ccPageLabel'
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPublicCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiPublicCache.BUILD_GROUP`

String that specifies the build group for the cache, which groups all cache records built at approximately the same time for simpler maintenance.

### `ccrz.ccApiPublicCache.BUILD_TIMESTAMP`

Datetime for the build group, which identifies when the cache record was built. For all cache records that belong to the same build group, use the same timestamp.

**ccrz.ccApiPublicCache.EFFECTIVE\_END**

Datetime when the cache is no longer valid. If you don't include this key in your input map, the method assigns a datetime 50 years from the current date and time.

**ccrz.ccApiPublicCache.EFFECTIVE\_START**

Datetime when the cache record becomes valid. If you don't include this key in your input map, the method uses the constant *NOW*, which specifies the current date and time.

**ccrz.ccApiPublicCache.ENABLE\_ON\_CLOSE**

Boolean

Value	Usage
<i>true</i>	Enable the cache record after the <code>ccrz.ccApiPublicCache.close</code> method is complete.
<i>false</i> (default)	The cache record remains disabled after it's created.

**ccrz.ccApiPublicCache.LOCALE**

String that specifies a locale to insert a new public cache record for. If you don't include this key in your input map, the new cache record applies to all locales.

**ccrz.ccApiPublicCache.PAGENAME**

String that specifies the name of a page to insert a new public cache record for. If you don't include this key in your input map, the new cache record applies to all pages.

**ccrz.ccApiPublicCache.STORE\_NAME**

String that specifies a storefront to insert a new public cache record for. If you don't include this key in your input map, the new cache record applies to all storefronts.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The call completed.
<i>false</i>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <i>false</i>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiPublicCache.CONTEXT**

`ccrz.ccApiPublicCache.CacheContext` object that you pass to `ccrz.ccApiPublicCache.appendTo` and `ccrz.ccApiPublicCache.close` method calls for this cache record.


## ccrz.ccApiPublicCache.appendTo

Adds content to a public cache record created with the `ccrz.ccApiPublicCache.startWrite` method. Call this method as often as necessary to populate the cache. This method returns a *cache context*, which you pass to subsequent `ccrz.ccApiPublicCache.appendTo` and `ccrz.ccApiPublicCache.close` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> appendTo (Map<String, Object>)
```

### Service Layer Classes

This method doesn't support service layer extensions or overrides.

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### **ccrz.ccApiPublicCache.CACHE\_NAME**

String that specifies the name of cached data to add, such as:

- `ccCategoryTree` for cached page label data that `ccrz.ccCategoryCacheBuildJob` builds.
- `ccPageLabel` for cached page label data that `ccrz.ccPageLabelCacheBuildJob` builds.

```
ccrz.ccApiPublicCache.CACHE_NAME => 'ccPageLabel'
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPublicCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiPublicCache.CONTEXT`

`ccrz.ccApiPublicCache.CacheContext` object returned from `ccrz.ccApiPublicCache.startWrite` or previous `ccrz.ccApiPublicCache.appendTo` method calls.

### `ccrz.ccApiPublicCache.CONTENT`

String that specifies the content to add to the cache record.

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### `ccrz.ccApiPublicCache.CONTEXT`

`ccrz.ccApiPublicCache.CacheContext` object that you pass to `ccrz.ccApiPublicCache.appendTo` and `ccrz.ccApiPublicCache.close` method calls for this cache record.

## `ccrz.ccApiPublicCache.close`


Stops writing any more content to the public cache record created with the `ccrz.ccApiPublicCache.startWrite` and `ccrz.ccApiPublicCache.appendTo` methods. If you passed `ccrz.ccApiPublicCache.ENABLE_ON_CLOSE => true` in the input key for `ccrz.ccApiPublicCache.startWrite`, the cache is enabled after `ccrz.ccApiPublicCache.close` completes.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> close(Map<String, Object>)
```

## Service Layer Classes


This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiPublicCache.CACHE_NAME`

String that specifies the name of cached data to close, such as:

- `ccCategoryTree` for cached page label data that `ccrz.ccCategoryCacheBuildJob` builds.
- `ccPageLabel` for cached page label data that `ccrz.ccPageLabelCacheBuildJob` builds.

```
ccrz.ccApiPublicCache.CACHE_NAME => 'ccPageLabel'
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPublicCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiPublicCache.CONTEXT`

`ccrz.ccApiPublicCache.CacheContext` object returned from `ccrz.ccApiPublicCache.appendTo` method calls.

## Outputs


Map<String, Object> that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .


## `ccrz.ccApiPublicCache.disableOld`

Disables cache records that have the same cache name, storefront, page, and locale as another cache record with a more recent Effective Start date and time. This method changes the value of the Enabled field on the affected public cache records. Make sure that you specify a cache name, storefront, page name, and locale in the method's input map.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> disableOld (Map<String, Object>)
```

## Service Layer Classes

This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiPublicCache.CACHE_NAME`

String that specifies the name of cached data to disable, such as:

- `ccCategoryTree` for cached page label data that `ccrz.ccCategoryCacheBuildJob` builds.
- `ccPageLabel` for cached page label data that `ccrz.ccPageLabelCacheBuildJob` builds.

```
ccrz.ccApiPublicCache.CACHE_NAME => 'ccPageLabel'
```



**Note:** If this key isn't specified, the method returns a `ccrz.ccApiPublicCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiPublicCache.LOCALE`

String that specifies a locale to disable old cache entries for. If locale doesn't apply to the cache entries you want to disable, pass an empty string:

```
ccrz.ccApiPublicCache.LOCALE => ''
```

### `ccrz.ccApiPublicCache.PAGENAME`

String that specifies a page to disable old cache entries for.

### `ccrz.ccApiPublicCache.STORE_NAME`

String that specifies a storefront to disable old cache entries for.

## Outputs

None

## `ccrz.ccApiPublicCache.toggleGroup`

Changes the value of the Enabled field for all public cache records that belong to the same build group.


## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> toggleGroup(Map<String, Object>)
```

## Service Layer Classes


This method doesn't support service layer extensions or overrides.

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiPublicCache.CACHE_NAME`

String that specifies a Cache Name value for the public cache records you want to update, such as:

- `ccCategoryTree` for cached page label data that `ccrz.ccCategoryCacheBuildJob` builds.
- `ccPageLabel` for cached page label data that `ccrz.ccPageLabelCacheBuildJob` builds.

```
ccrz.ccApiPublicCache.CACHE_NAME => 'ccPageLabel'
```

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApiPublicCache.NoCacheNameException`.

## Inputs (Optional)

The input map can also include the following keys:


### `ccrz.ccApiPublicCache.BUILD_GROUP`

String that specifies a Build Group value for the public cache records you want to update. This method updates all public cache records with the same `ccrz.ccApiPublicCache.CACHE_NAME` and `ccrz.ccApiPublicCache.BUILD_GROUP` values.

### `ccrz.ccApiPublicCache.SET_ENABLED`

Boolean

Value	Usage
<i>true</i>	Enable the Enabled field on each affected public cache record.
<i>false</i>	Disable the Enabled field on each affected public cache record.

 **Note:** If you don't include this key, by default this method changes the current value of the Enabled field to the opposite value. For example, if Enabled is true, this method changes Enabled to false.

## Outputs

None

## ccrz.ccApiRelatedProduct

Query related product records, which define relationships between products, such as accessories or competitive or alternative SKUs.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

IN THIS SECTION:

[fetch](#)

Executes a SOQL query of related product records and returns a list of records that match your query.

SEE ALSO:

[Create Relationships Between Products](#)


## ccrz.ccApiRelatedProduct.fetch

Executes a SOQL query of related product records and returns a list of records that match your query.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider


```
ccrz.ccServiceRelatedProduct
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### **ccrz.ccApiRelatedProduct.PARAM\_BY\_ASC**

Boolean

Value	Usage
<i>true</i>	Sort returned related product records by Sequence value in ascending order. For example, sort from 100 to 900.
<i>false</i> (default)	Sort returned products by Sequence in descending order. For example, sort from 900 to 100.

**ccrz.ccApiRelatedProduct.PARAM\_BY\_NULLS\_LAST**

Boolean

Value	Usage
<i>true</i>	Place all related product records with a null Sequence value at <a href="#">the end of the returned related product data</a> .
<i>false</i> (default)	Place all related product records with a null Sequence value at the beginning of the returned related product data.

**ccrz.ccApiRelatedProduct.PRODUCTID**

String that specifies a product ID whose related products you want to query.

**ccrz.ccApiRelatedProduct.PRODUCTIDLIST**

Set&lt;String&gt; of specific product IDs whose related products you want to query.

```
new Set<String>{'Product_ID_1', 'Product_ID_2'}
```

**ccrz.ccApiRelatedProduct.RELATED\_TYPE**

Set&lt;String&gt; of specific Related Product Type field values you want to filter by.

```
new Set<String>{'Accessories', 'UpSell'}
```

**ccrz.ccApiRelatedProduct.RELATEDPRODUCTID**

String that specifies a related product record's ID.

**ccrz.ccApiRelatedProduct.RELATEDPRODUCTIDLIST**

Set&lt;String&gt; of specific related product record IDs to query.

```
new Set<String>{'Related_Product_ID_1', 'Related_Product_ID_2'}
```

 **Note:** This method evaluates the ID keys in the following order of precedence:

1. `ccrz.ccApiRelatedProduct.RELATEDPRODUCTID`
2. `ccrz.ccApiRelatedProduct.RELATEDPRODUCTIDLIST`
3. `ccrz.ccApiRelatedProduct.PRODUCTID`
4. `ccrz.ccApiRelatedProduct.PRODUCTIDLIST`

## Outputs


Map&lt;String, Object&gt; that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint.</a>

**ccrz.ccApiRelatedProduct.RELATEDPRODUCTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [related product record](#).

**ccrz.ccApiProduct.PRODUCTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [product record](#) referenced from one of the related product records included in `ccrz.ccApiRelatedProduct.RELATEDPRODUCTLIST`.

## ccrz.ccApiRichContent

Query rich content records used for custom storefront content.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

#### [fetch](#)

Executes a SOQL query of rich content records, and returns a list of rich content records that match your query.

### SEE ALSO:

[Salesforce Help: B2B Commerce Content Authoring](#)


## ccrz.ccApiRichContent.fetch

Executes a SOQL query of rich content records, and returns a list of rich content records that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider

```
ccrz.ccServiceRichContent
```

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

#### **ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

**ccrz.ccApiRichContent.IDS**

Set<String> of specific rich content IDs to query:

```
new Set<String>{'Rich_Content_ID_1', 'Rich_Content_ID_2'}
```

**ccrz.ccApiRichContent.KEYS**

Set<String> of specific Content Key field values to filter rich content records by.

```
new Set<String>{'Content_Key_1', 'Content_Key_2'}
```

**ccrz.ccApiRichContent.LOCALE**

String that specifies a locale to return localized rich content for in the `ccrz.ccApiRichContent.RICH_CONTENT_LIST` return key. The service layer translates the values from `ccrz__E_RichContentI18N__c` records and handles fallback languages, if necessary.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccApiRichContent.STATUS**

String that specifies a Status field value to filter rich content records by.

**ccrz.ccApiRichContent.STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiRichContent.RICH\_CONTENT\_LIST**

List<Map<String, Object>>, where each Map<String, Object> represents a rich content record.

 **Example: Example**

Query rich content records by passing in a specific Content Key value.

```
Set<String> contentKeys = new Set<String>{'Content_Key'};
```

```

Map<String, Object> inputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiRichContent.KEYS => contentKeys,
    ccrz.ccApi.SIZING => new Map<String, Object>{
        ccrz.ccApiRichContent.ENTITYNAME => new Map<String, Object>{
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
        }
    }
};

Map<String, Object> outputData = ccrz.ccApiRichContent.fetch(inputData);

Boolean success = (Boolean)outputData.get(ccrz.ccApi.SUCCESS);

if(success) {
    List<Map<String, Object>> richContents = (List<Map<String,
Object>>)outputData.get(ccrz.ccApiRichContent.RICH_CONTENT_LIST);
    System.debug('richContents=' + JSON.serializePretty(richContents));
} else {
    List<ccrz.cc_bean_Message> messages =
(List<ccrz.cc_bean_Message>)outputData.get(ccrz.ccApi.MESSAGES);
    System.debug('messages=' + messages);
}

```

## ccrz.ccApiSeller

Query seller records by IDs or by geographic location.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

IN THIS SECTION:

#### [fetch](#)

Executes a SOQL query of seller records, and returns a list of sellers that match your query.



[search](#)

Executes a SOSL query of seller records, and returns the IDs of sellers that match the search query.

SEE ALSO:

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)


## ccrz.ccApiSeller.fetch

Executes a SOQL query of seller records, and returns a list of sellers that match your query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch(Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider


`ccrz.ccServiceSeller`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### **ccrz.ccApiSeller.PARAM\_BY\_ASC**

Boolean

Value	Usage
<i>true</i>	Sort returned sellers by name in alphabetical order.
<i>false</i>	Sort returned sellers by name in reverse alphabetical order.

### **ccrz.ccApiSeller.PARAM\_BY\_DIST**

Boolean

Value	Usage
<i>true</i>	Sort returned sellers by distance, from nearest to farthest.
<i>false</i>	Sort returned sellers by name.

### **ccrz.ccApiSeller.PARAM\_ORIGIN\_LAT**

Decimal that specifies a seller location's latitude value to query by.

### **ccrz.ccApiSeller.PARAM\_ORIGIN\_LNG**

Decimal that specifies a seller location's longitude value to query by.

### **ccrz.ccApiSeller.PARAM\_SEARCH\_DIST**

Decimal that specifies a radius around the `ccrz.ccApiSeller.PARAM_ORIGIN_LAT` and `ccrz.ccApiSeller.PARAM_ORIGIN_LNG` location values.

### **ccrz.ccApiSeller.PARAM\_SEARCH\_UNIT**

String that specifies the units of the `ccrz.ccApiSeller.PARAM_SEARCH_DIST` location radius, such as *mi* or *km*.

### **ccrz.ccApiSeller.SELLERID**

String that specifies a seller ID to query.

### **ccrz.ccApiSeller.SELLERIDLIST**

Set<String> of specific seller IDs to query.

```
new Set<String>{'Seller_ID_1', 'Seller_ID_2'}
```

### **ccrz.ccApiSeller.SELLERLOCALE**

String that specifies a locale that you want to return localized seller values for in the `ccrz.ccApiSeller.SELLERLIST` return key. The service layer translates the values from [ccrz\\_\\_E\\_SellerI18N\\_\\_c records](#) and handles fallback languages, if necessary.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

## Outputs


Map<String, Object> that can include the following keys:

### ccrz.ccApi.API\_VERSION

Integer that indicates which API version was used for the query.

### ccrz.ccApi.SUCCESS

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is false, <a href="#">consider rolling back the API transaction to a previous savepoint.</a>

### ccrz.ccApiSeller.SELLERLIST

List<Map<String, Object>>, where each Map<String, Object> represents a [seller record](#).

### Example: Example

Pass in a pair of seller IDs and capture the seller names in a string variable.

```
Set<String> sellerIdList = new Set<String>{'Seller_ID_1', 'Seller_ID_2'};

Map<String, Object> inputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiSeller.SELLERIDLIST => sellerIdList,
    ccrz.ccApiSeller.SELLERLOCALE => 'fr_CA',
    ccrz.ccApi.SIZING => new Map<String, Object>{
        ccrz.ccApiSeller.ENTITYNAME => new Map<String, Object>{
            ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S
        }
    }
};

try {
    Map<String, Object> outputData = ccrz.ccApiSeller.fetch(inputData);
    if (outputData.get(ccrz.ccApiSeller.SELLERLIST) != null) {
        // The cast to List<Map<String, Object>> is necessary...
        List<Map<String, Object>> outputSellerList = (List<Map<String, Object>>)
outputData.get(ccrz.ccApiSeller.SELLERLIST);

        // The cast, again, is necessary...
        String sellerName = (String) outputSellerList[0].get('sfdcName');
    }
} catch (Exception e) {
```

```
// Error handling...
}
```

SEE ALSO:

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)


## ccrz.ccApiSeller.search

Executes a SOSL query of seller records, and returns the IDs of sellers that match the search query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> search (Map<String, Object>)
```

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApiSeller.PARAM\_SEARCH\_DIST**

Decimal that specifies a radius around the `ccrz.ccApiSeller.PARAM_SEARCH_LAT` and `ccrz.ccApiSeller.PARAM_SEARCH_LNG` location values.

**ccrz.ccApiSeller.PARAM\_SEARCH\_LAT**

Decimal that specifies a seller location's latitude value to search by.

**ccrz.ccApiSeller.PARAM\_SEARCH\_LNG**

Decimal that specifies a seller location's longitude value to search by.

**ccrz.ccApiSeller.PARAM\_SEARCH\_UNIT**

String that specifies the units of the `ccrz.ccApiSeller.PARAM_SEARCH_DIST` location radius, such as *mi* or *km*.

**ccrz.ccApiSeller.SELLERLOCALE**

String that specifies a locale that you want to return localized seller values for in the `ccrz.ccApiSeller.SELLERLIST` return key. The service layer translates the values from `ccrz__E_SellerI18N__c` records and handles fallback languages, if necessary.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccService.SEARCHSTRING**

String to search for from all text fields on `ccrz__E_Seller__c` and `ccrz__E_SellerI18N__c` records, including name and description fields. You can refine the search in the following ways:

- To find all sellers within a specific location—instead of searching sellers for a specific string—pass *all* for `ccrz.ccService.SEARCHSTRING`. Include the latitude, longitude, distance, and unit location keys.
- To search all sellers for a certain string—regardless of location—pass a specific value for `ccrz.ccService.SEARCHSTRING` and don't include any location keys.
- To search all sellers within a specific location for a certain string, pass a specific value for `ccrz.ccService.SEARCHSTRING` and include the location keys.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, consider rolling back the API transaction to a previous savepoint.</p>

**ccrz.ccService.SEARCHRESULTS**

Set<String> of seller IDs that match for your search criteria.

### Example: Example

Find all sellers within a half-mile radius of a specific location.

```
Map<String, Object> inputData = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccService.SEARCHSTRING => 'all',
    ccrz.ccApiSeller.SELLERLOCALE => 'fr_CA',
    ccrz.ccApiSeller.PARAM_SEARCH_LAT => 41.8835932,
    ccrz.ccApiSeller.PARAM_SEARCH_LNG => -87.6385858,
    ccrz.ccApiSeller.PARAM_SEARCH_DIST => 0.5,
    ccrz.ccApiSeller.PARAM_SEARCH_UNIT => 'mi'
};

try {
    Map<String, Object> outputData = ccrz.ccApiSeller.search(inputData);
    Set<String> searchResults = (Set<String>)
outputData.get(ccrz.ccService.SEARCHRESULTS);

} catch (Exception e) {
    System.debug('MMM--exception: ' + e);
}
```

SEE ALSO:

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccApiSpec

Query specification definitions and product spec index records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

#### [fetch](#)

Executes a SOQL query of spec records, and returns a list of spec definitions that match your query.

#### [ccrz.ccApiSpec.fetchProductSpecIndices](#)

Executes a SOQL query of product spec index records, which define a single instance of a spec value for a product.

### SEE ALSO:

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)


## ccrz.ccApiSpec.fetch

Executes a SOQL query of spec records, and returns a list of spec definitions that match your query.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceSpec`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

**ccrz.ccApiSpec.ISCOMPARABLE**

Boolean

Value	Usage
<code>true</code>	Return only specs where the <code>ccrz__IsComparable__c</code> field is true.
<code>false</code> (default)	Ignore the value of the <code>ccrz__IsComparable__c</code> field when querying spec records.

**ccrz.ccApiSpec.PARAM\_BY\_ASC**

Boolean

Value	Usage
<code>true</code>	Sort returned specs by ID in ascending order. For example, sort from 100 to 900. If you pass <code>ccrz.ccApiSpec.PARAM_BY_SEQ =&gt; true</code> , specs sort by Sequence value in ascending order.
<code>false</code> (default)	Sort returned specs by ID in descending order.

**ccrz.ccApiSpec.PARAM\_BY\_NULLS\_LAST**

Boolean

Value	Usage
<code>true</code>	Place all specs with a null Sequence value at <a href="#">the end of the returned spec data</a> .
<code>false</code> (default)	Place all specs with a null Sequence value at the beginning of the returned spec data.

**ccrz.ccApiSpec.PARAM\_BY\_SEQ**

Boolean

Value	Usage
<code>true</code>	Sort returned specs by Sequence value in descending order. For example, sort from 900 to 100.



Value	Usage
<i>false</i> (default)	Sort returned specs by ID in descending order.

**ccrz.ccApiSpec.SPECCATALOGVISIBLE**

Boolean

Value	Usage
<i>true</i>	Return only specs where the <code>ccrz__IsVisibleInCatalog__c</code> field is true.
<i>false</i> (default)	Ignore the value of the <code>ccrz__IsVisibleInCatalog__c</code> field when querying spec records.

**ccrz.ccApiSpec.SPECFILTER**

Boolean

Value	Usage
<i>true</i>	Return only specs where the <code>ccrz__UseForFilter__c</code> field is true.
<i>false</i> (default)	Ignore the value of the <code>ccrz__UseForFilter__c</code> field when querying spec records.

**ccrz.ccApiSpec.SPECID**

String that specifies the ID of a spec to query.

**ccrz.ccApiSpec.SPECIDLIST**

Set&lt;String&gt; of spec IDs whose data you want to return:

```
new Set<String>{ 'Spec_ID_1', 'Spec_ID_2' }
```

**ccrz.ccApiSpec.SPECLOCALE**

String that specifies a locale that you want to return localized spec records for in the `ccrz.ccApiSpec.SPECLIST` return key. The service layer translates the values from spec records created for that locale and handles fallback languages, if necessary.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

## Outputs

Map<String, Object> that can include the following keys:


**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**


Boolean

Value	Usage
<i>true</i>	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### `ccrz.ccApiSpec.SPECLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a [spec record](#).

 **Tip:** To specify which fields the API returns for each cart record, [request a specific data size](#) for the `ccrz.ccApiSpec.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiSpec.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields are available with different data sizes, see the [ccrz\\_\\_E\\_Spec\\_\\_c](#) reference.

### Example: Examples

Fetch data for a pair of spec IDs, and parse the returned list for the first spec's name.

```
Map<String, Object> specFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiSpec.SPECIDLIST => {'Spec_ID_1', 'Spec_ID_2'}
};

try {
    Map<String, Object> specReturnData = ccrz.ccApiSpec.fetch(specFetchQuery);
    if (specReturnData.get(ccrz.ccApiSpec.SPECLIST) != null) {
        List<Map<String, Object>> specList = (List<Map<String, Object>>)
specReturnData.get(ccrz.ccApiSpec.SPECLIST);
        String specName = (String) specList[0].get('sfdcName');
    }
} catch (Exception e) {
    // Error handling...
}
```

Fetch all spec records and sort the return data so that records with null Sequence values are last.

```
Map<String, Object> specFetchQuery = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiSpec.PARAM_BY_SEQ => true,
    ccrz.ccApiSpec.PARAM_BY_ASC => false,
    ccrz.ccApiSpec.PARAM_BY_NULLS_LAST => true
};

try {
    Map<String, Object> specReturnData = ccrz.ccApiSpec.fetch(specFetchQuery);
    if (specReturnData.get(ccrz.ccApiSpec.SPECLIST) != null) {
```

```

        List<Map<String, Object>> specList = (List<Map<String, Object>>)
specReturnData.get(ccrz.ccApiSpec.SPECLIST);
        for (Map<String, Object> spec : specList) {
            System.debug('***** seq: ' + spec.get('sequence'));
        }
    }
} catch (Exception e) {
    // Error handling...
}

```

SEE ALSO:

[ccrz\\_\\_E\\_Spec\\_\\_c](#)


## ccrz.ccApiSpec.fetchProductSpecIndices

Executes a SOQL query of product spec index records, which define a single instance of a spec value for a product.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetchProductSpecIndices (Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider

```
ccrz.ccServiceProductSpecIndex
```

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

**Inputs (Optional)**

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that specifies the [scope of data that this method returns](#). To specify which fields are available in the return data for each product spec index record, add `ccrz.ccApiSpec.PSI_ENTITYNAME` as a key in this map:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiSpec.PSI_ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

**ccrz.ccApiSpec.PRODUCT\_IDS**


`Set<String>` of product IDs whose product spec index records you want to return:

```
new Set<String>{'Product_ID_1', 'Product_ID_2'}
```

**ccrz.ccApiSpec.PRODUCT\_SPEC\_INDICES**

`List<Map<String, Object>>`, where each `Map<String, Object>` specifies a spec ID and a particular spec value whose corresponding product spec index records you want to return, such as:

```
[ {
    "specValues" : [ {
        "value" : "Blue"
    } ],
    "sfid" : "alh6A00000045s5QAA"
} ]
```

 **Note:** If you specify multiple values for the same spec, the method applies an OR condition for evaluating all values.

For a spec where Filter Type is Slider, you can specify a `filterMin` and `filterMax` range for filtering:

```
[ {
    "sfid" : "alh6A00000045s5QAA",
    "filterMin" : "1.0",
    "filterMax" : "10.0"
} ]
```

**ccrz.ccApiSpec.SPECIDLIST**

`Set<String>` of product spec index IDs whose data you want to return:

```
new Set<String>{'Product_Spec_Index_ID_1', 'Product_Spec_Index_ID_2'}
```

**ccrz.ccApiSpec.SPECLOCALE**

String that specifies a locale. The method returns only the product spec index records where the referenced spec has the specified locale.

**ccrz.ccApiSpec.USE\_FOR\_FILTER**

Boolean

Value	Usage
<code>true</code>	Return only product spec index records where the related spec's <code>ccrz__UseForFilter__c</code> field is true.
<code>false</code> (default)	Ignore the value of a related spec's <code>ccrz__UseForFilter__c</code> field when querying product spec index records.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean


Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiSpec.PRODUCT\_SPEC\_INDICES**

Data from the [product spec index records](#) that match the fetch query. This return data can take the following types, depending on values specified in the input map:

Type	Input Map Criteria
Map<String, Map<String, Object>>	Default return format
List<Map<String, Object>>	<pre>ccrz.ccApi.SIZING =&gt; new Map&lt;String, Object&gt; {     ccrz.ccApiSpec.PSI_ENTITYNAME =&gt; new Map&lt;String, Object&gt; {         ccrz.ccApi.SZ_DATA =&gt;         ccrz.ccApi.SZ_GROUPBY     } }</pre>
List< <a href="#">ccrz__E_ProductSpecIndex__c</a> >	<pre>ccrz.ccApi.SIZING =&gt; new Map&lt;String, Object&gt; {</pre>

Type	Input Map Criteria
	<pre>ccrz.ccApiSpec.PSI_ENTITYNAME =&gt; new Map&lt;String, Object&gt; {     ccrz.ccApi.SZ_SKIPTRZ =&gt; true }</pre>
List<AggregateResult>	<pre>ccrz.ccApi.SIZING =&gt; new Map&lt;String, Object&gt; {     ccrz.ccApiSpec.PSI_ENTITYNAME =&gt; new Map&lt;String, Object&gt; {     ccrz.ccApi.SZ_DATA =&gt; ccrz.ccApi.SZ_GROUPBY,     ccrz.ccApi.SZ_SKIPTRZ =&gt; true     } }</pre>

 **Tip:** To specify which fields the API returns for each product spec index record, [request a specific data size](#) for the `ccrz.ccApiSpec.PSI_ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiSpec.PSI_ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields are available with different data sizes, see the [ccrz\\_\\_E\\_ProductSpecIndex\\_\\_c](#) reference.

SEE ALSO:

[ccrz\\_\\_E\\_Spec\\_\\_c](#)

[ccrz\\_\\_E\\_ProductSpec\\_\\_c](#)

[ccrz\\_\\_E\\_ProductSpecIndex\\_\\_c](#)

## ccrz.ccApiSiteIndex

Query, create, update, or remove site index records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

#### [create](#)

Creates site index records for a storefront, based on categories and locales that you specify.

#### [fetch](#)

Executes a SOQL query of site index records, and returns a list of records that match you query.

#### [mapUrl](#)

Returns the category or product default URL that corresponds to a specified friendly URL.

#### [remove](#)

Delete site index records. B2B Commerce for Visualforce uses this method in site index batch jobs.

#### [revise](#)

Updates the status of the Active field on existing site index records. B2B Commerce for Visualforce uses this method in site index batch jobs.

### SEE ALSO:

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

[ccrz\\_\\_E\\_SiteIndex\\_\\_c](#)


## ccrz.ccApiSiteIndex.create

Creates site index records for a storefront, based on categories and locales that you specify.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> create (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


[ccrz.ccLogicSICreate](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:


### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.


### `ccrz.ccApiSiteIndex.CATEGORY_IDS`

`Set<String>` of category IDs to create site index records for.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.


### `ccrz.ccApiSiteIndex.LOCALES`

`Set<String>` of locales to create site index records for. Typically, you want to create site index records for all the locales that your storefront supports.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.


### `ccrz.ccApiSiteIndex.ROOT_CATEGORY_ID`

String that specifies the ID of the [storefront's root category](#).

 **Note:** If this key isn't specified, the method doesn't return an exception. But, this key ensures that the created site index records specify the correct storefront and reflect the storefront's category tree.

### `ccrz.ccApiSiteIndex.STOREFRONT`

String that specifies the name of the storefront to create site index records for.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:



**ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiSiteIndex.fetch` for returning the created site index records.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

**Outputs**


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiSiteIndex.IDS**

Set<String> of IDs of the new `ccrz__E_SiteIndex__c` records.

 **Example: Example**

Create site index records for a pair of category IDs and two locales.

```
Map<String, Object> siteIndexesToCreate = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiSiteIndex.CATEGORY_IDS => new Set<String>{'Category_ID_1',
    'Category_ID_2'},
    ccrz.ccApiSiteIndex.STOREFRONT => 'storefrontName',
    ccrz.ccApiSiteIndex.LOCALES => new Set<String>{'en_US', 'ja'},
    ccrz.ccApiSiteIndex.ROOT_CATEGORY_ID => 'Root_Category_ID'
};

try {
    Map<String, Object> createdSiteIndexes =
    ccrz.ccApiSiteIndex.create(siteIndexesToCreate);
} catch (Exception e) {
```

```
// Error handling...
}
```

SEE ALSO:

[ccrz\\_\\_E\\_SiteIndex\\_\\_c](#)

[ccrz.ccSiteIndexBuilder](#)

[Translate Storefront Text for Buyers in Different Locales](#)

## ccrz.ccApiSiteIndex.fetch

Executes a SOQL query of site index records, and returns a list of records that match you query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider


`ccrz.ccServiceSiteIndex`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

Map<String, Object> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

### `ccrz.ccApiSiteIndex.ACTIVE`

Boolean

Value	Usage
<code>true</code>	Return only site index records where the Active field is enabled.
<code>false</code>	Ignore the value of the Active field when querying site index records.

### `ccrz.ccApiSiteIndex.CATEGORY_IDS`

Set<String> of category IDs whose corresponding site index records you want to query.

### `ccrz.ccApiSiteIndex.IDS`

Set<String> of site index IDs to query.

### `ccrz.ccApiSiteIndex.LOCALE`

String that specifies a locale to query site index records for.

### `ccrz.ccApiSiteIndex.LOCALES`

Set<String> of locales to query site index records for.

### `ccrz.ccApiSiteIndex.STOREFRONT`

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### `ccrz.ccApiSiteIndex.URL_HASH`

String that specifies a category's hashed friendly URL value to query site index records for. This string matches the value in a site index record's `Friendly URL Hash` field.

## Outputs

Map<String, Object> that can include the following keys:


### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`


Boolean

Value	Usage
<code>true</code>	The call completed.

Value	Usage
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint.</a>

### `ccrz.ccApiSiteIndex.SITE_INDICES`

`Map<String, Object>` of `ccrz__E_SiteIndex__c` records that match the query.

 **Tip:** To specify which fields the API returns for each site index record, [request a specific data size](#) for the `ccrz.ccApiSiteIndex.ENTITYNAME` key in your input data:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiSiteIndex.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M
    }
}
```

For information about which fields are available with different data sizes, see the `ccrz__E_SiteIndex__c` reference.

### Example: Example

Query all site index records for a specific storefront and locale.

```
Map<String, Object> siteIndexFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiSiteIndex.STOREFRONT => 'DefaultStore',
    ccrz.ccApiSiteIndex.LOCALE => 'en_US'
};

try {
    Map<String, Object> siteIndexFetchData =
ccrz.ccApiSiteIndex.fetch(siteIndexFetchQuery);
    if (siteIndexFetchData.get(ccrz.ccApiSiteIndex.SITE_INDICES) != null) {
        Map<String, Object> siteIndexResults = (Map<String,
Object>)siteIndexFetchData.get(ccrz.ccApiSiteIndex.SITE_INDICES);
        // ...
    }
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

[ccrz\\_\\_E\\_SiteIndex\\_\\_c](#)


## `ccrz.ccApiSiteIndex.mapUrl`

Returns the category or product default URL that corresponds to a specified friendly URL.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> mapUrl (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicSMapUrl`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiSiteIndex.FULL_URL`

String that specifies the friendly URL for a product or category whose default URL you want to return. This value can be a URL that the `ccrz.ccApiProduct.getUrlFor` or `ccrz.ccApiCategory.getUrlFor` method returns.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiSiteIndex.STOREFRONT`

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

#### `ccrz.ccApiSiteIndex.URL_LOCALE`

String that specifies one of your storefront's supported locales.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

## Outputs


Map<String, Object> that can include the following keys:

#### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

#### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

#### `ccrz.ccApiSiteIndex.MAPPED_URL`

String that specifies the category or product default URL that corresponds to the specified friendly URL.

### Example: Example

For a product's friendly URL, such as `/Coffee/Medium/MediumRoast`, return the default URL, such as `/ProductDetails?sku=MRC-1`.

```
inputFriendlyUrl = '/Coffee/Medium/MediumRoast';
outputDefaultUrl = '';

try {
    Map<String, Object> mapUrlReturnData = ccrz.ccApiSiteIndex.mapUrl(new Map<String,
    Object> {
        ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
        ccrz.ccApiSiteIndex.FULL_URL => inputFriendlyUrl
    });

    if (ccrz.ccUtil.isKeyValued(mapUrlReturnData, ccrz.ccApiSiteIndex.MAPPED_URL)) {
        outputDefaultUrl = (String)mapUrlReturnData.get(ccrz.ccApiSiteIndex.MAPPED_URL);
    }
} catch (Exception e) {
```

```
// handle exceptions
}
```

**SEE ALSO:**

[ccrz.ccApiCategory.getUrlFor](#)

[ccrz.ccApiProduct.getUrlFor](#)

[Salesforce Help Specify SEO Metadata for Products and Categories](#)

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)


## ccrz.ccApiSiteIndex.remove

Delete site index records. B2B Commerce for Visualforce uses this method in site index batch jobs.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> remove (Map<String, Object>)
```

### Service Layer Classes

**Logic Service Provider**

[ccrz.ccLogicSIRemove](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:


**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

**ccrz.ccApiSiteIndex.IDS**

`List<String>` of site index IDs to remove.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

## Example: Example

Delete site index records whose IDs you specify.

```
Map<String, Object> siteIndexesToRemove = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiSiteIndex.IDS => new List<String>{'Site_Index_ID_1', 'Site_Index_ID_2'}
};

try {
    Map<String, Object> siteIndexRemoveResult =
    ccrz.ccApiSiteIndex.remove(siteIndexesToRemove);
} catch (Exception e) {
```



```
// Error handling...
}
```

SEE ALSO:

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

[ccrz\\_\\_E\\_SiteIndex\\_\\_c](#)

[ccrz.ccSiteIndexBuilder](#)

## ccrz.ccApiSiteIndex.revise

Updates the status of the Active field on existing site index records. B2B Commerce for Visualforce uses this method in site index batch jobs.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.



**Warning:** This method doesn't allow a guest user to update records. If you're creating subscriber code that requires a guest user to update records, you have the following options:

- If you're using B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11), invoke the [ccrz.ccLogic.updateObjects](#) utility method.
- Execute the DML operation in your custom Apex class using the [without sharing](#) accessor.

### Signature

```
global static Map<String, Object> revise (Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider

[ccrz.ccServiceSiteIndex](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiSiteIndex.SITE_INDICES`

`Map<String, ccrz__E_SiteIndex__c>` of the site index records to update.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiSiteIndex.fetch` for returning the updated site index records.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.


### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiSiteIndex.SITE_INDICES`

`Map<String, Object>` of the updated `ccrz__E_SiteIndex__c` records.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

SEE ALSO:

[ccrz\\_\\_E\\_SiteIndex\\_\\_c](#)  
[ccrz.ccSiteIndexBuilder](#)

## ccrz.ccApiStoredPayment

Query, create, update, and delete payments saved in the My Wallet section of the My Account page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

IN THIS SECTION:

[create](#)

Creates a stored payment record from the fields and values that you specify.

[fetch](#)

Executes a SOQL query of stored payment records, and returns a list of records that match you query.

[remove](#)

Deletes stored payment records.

[revise](#)

Updates certain fields on a stored payment record.

[Extension Examples for Stored Payments](#)

To implement custom business logic for stored payments, create custom subscriber code that extends the default data service provider and logic service provider classes. The following code examples provide starting points for working around limits in the default stored payment functionality and defining custom requirements.

SEE ALSO:

[Salesforce Help Allow a Buyer to Manage Stored Payments on the My Account Page](#)


## ccrz.ccApiStoredPayment.create

Creates a stored payment record from the fields and values that you specify.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> create (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider


[ccrz.ccLogicStoredPaymentCreate](#)

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### `ccrz.ccApiStoredPayment.STORED_PAYMENTS_LIST`

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the `ccrz__E_StoredPayment__c` record to create. You can specify the following fields:

account	endDate	startDate
accountNumber	expMonth	storefront
accountType	expYear	subaccountNumber

displayName	paymentType	token
enabled	sequence	user

 **Note:** By default, this method creates only one cart record at one time. Even though this key specifies a list, the method evaluates only the first item in the list.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiStoredPayment.fetch` for returning the created stored payment.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiStoredPayment.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean


Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### `ccrz.ccApiStoredPayment.ID`

ID of the new `ccrz__E_StoredPayment__c` record.

### `ccrz.ccApiStoredPayment.STOREDPAyMENTSList`

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the created `ccrz__E_StoredPayment__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

SEE ALSO:

[ccrz.ccApiStoredPayment.fetch](#)


## ccrz.ccApiStoredPayment.fetch

Executes a SOQL query of stored payment records, and returns a list of records that match you query.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider

`ccrz.ccServiceStoredPayment`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies [options for how the method returns data](#).

 **Note:** Specific data size requests don't apply to this method. The same fields are always returned for each object, regardless of any specific data size request in your input map.

### **ccrz.ccApiStoredPayment.ACCOUNTTYPES**

List<String> of specific `ccrz__AccountType__c` values on `ccrz__E_StoredPayment__c` records to query, such as purchase order (PO), credit card, and so on. For more information about payment types, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_payment\\_types.htm](https://help.salesforce.com/articleView?id=b2b_commerce_payment_types.htm).

 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

### **ccrz.ccApiStoredPayment.BYASC**

Boolean

Value	Usage
<i>true</i>	Sort the returned stored payment records in ascending order.
<i>false</i> (default)	Sort the returned stored payment records in descending order.

### **ccrz.ccApiStoredPayment.BYUSER**

String that specifies the Salesforce ID of a user. This method returns only the `ccrz__E_StoredPayment__c` records where the `ccrz__User__c` field references this user ID.

 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

### **ccrz.ccApiStoredPayment.EFFACT**

String that specifies the Salesforce ID of an effective account for filtering the query.

### **ccrz.ccApiStoredPayment.ID**

String that specifies the Salesforce ID of the stored payment record to query.

### **ccrz.ccApiStoredPayment.IDS**

Set<String> that specifies the Salesforce IDs of the stored payment records to query.

### **ccrz.ccApiStoredPayment.ISEENABLED**


Boolean

Value	Usage
<i>true</i>	Return only stored payment records where the <code>ccrz__Enabled__c</code> field is <i>true</i> .
<i>false</i> (default)	Ignore the value of the <code>ccrz__Enabled__c</code> field.

### **ccrz.ccApiStoredPayment.ISREADONLY**

Boolean

Value	Usage
<code>true</code>	Return only stored payment records where the <code>ccrz__ReadOnly__c</code> field is <code>true</code> .
<code>false</code> (default)	Ignore the value of the <code>ccrz__ReadOnly__c</code> field.

 **Note:** This input is available only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

#### **ccrz.ccApiStoredPayment.STOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs


Map<String, Object> that can include the following keys:

#### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


#### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .


#### **ccrz.ccApiStoredPayment.STOREDPAymentLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_StoredPayment__c` record.

 **Note:** Each stored payment record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.

#### **ccrz.ccApiSubscription.SUBSCRIPTIONS**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_Subscription__c` record associated with a returned `ccrz__E_StoredPayment__c` record.

 **Note:** This output is returned only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12), and only when the input data specifies `ccrz.ccApi.SZ_ASSOC => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiStoredPayment.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_ASSOC => true
    }
}
```



```

    }
}

```

### Example: Example

Fetch stored payment records by ID, and capture the first result's method of payment.

```

Map<String, Object> storedPaymentQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiStoredPayment.IDS => new Set<String>{'stored_payment_ID_1',
'stored_payment_ID_2'}
};

try {
    Map<String, Object> storedPaymentResults =
ccrz.ccApiStoredPayment.fetch(storedPaymentQuery);
    if (storedPaymentResults.get(ccrz.ccApiStoredPayment.STORED_PAYMENT_LIST) != null)
    {
        List<Map<String, Object>> outputPaymentsList = (List<Map<String, Object>>)
storedPaymentResults.get(ccrz.ccApiStoredPayment.STORED_PAYMENT_LIST);
        String paymentTypes = (String) outputPaymentsList[0].get('accountType');
    }
} catch (Exception e) {
    // Error handling...
}

```

SEE ALSO:

[Salesforce Help Set Up Payment Types for Storefront Purchases](#)


## ccrz.ccApiStoredPayment.remove

Deletes stored payment records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> remove (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

```
ccrz.ccLogicStoredPaymentRemove
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

Include either of the following keys, but not both. If you don't include either key, the method returns a `ccrz.ccApi.MissingInputException`.

### **ccrz.ccApiStoredPayment.ID**

String that specifies the ID of a stored payment to remove.

### **ccrz.ccApiStoredPayment.IDS**

`List<String>` that specifies the IDs of stored payments to remove.

## Outputs


`Map<String, Object>` that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### **ccrz.ccApiSubscription.SUBSCRIPTIONS**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_Subscription__c` record associated with a removed `ccrz__E_StoredPayment__c` record.

 **Note:** This output is returned only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12), and only when the input data specifies `ccrz.ccApi.SZ_ASSOC => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiStoredPayment.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_ASSOC => true
    }
}
```


## ccrz.ccApiStoredPayment.revise

Updates certain fields on a stored payment record.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> revise (Map<String, Object>)
```

### Service Layer Classes

#### Data Service Provider

`ccrz.ccServiceStoredPayment`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiStoredPayment.fetch` for returning the updated stored payment.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiStoredPayment.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

### **ccrz.ccApiStoredPayment.STOREDPAyMENTSList**

List<Map<String, Object>>, where the Map<String, Object> represents the `ccrz__E_StoredPayment__c` record to update. You can specify the following fields:

account	endDate	startDate
accountNumber	expMonth	storefront
accountType	expYear	subaccountNumber
displayName	paymentType	token
enabled	sequence	user

## Outputs


Map<String, Object> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.


### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### **ccrz.ccApiStoredPayment.STOREDPAyMENTSList**

List<Map<String, Object>>, where the Map<String, Object> represents the updated `ccrz__E_StoredPayment__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

SEE ALSO:

[ccrz.ccApiStoredPayment.fetch](#)

## Extension Examples for Stored Payments

To implement custom business logic for stored payments, create custom subscriber code that extends the default data service provider and logic service provider classes. The following code examples provide starting points for working around limits in the default stored payment functionality and defining custom requirements.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Important:

- The following examples are for demo purposes only, and don't provide a complete extension solution. Often, extension is a more complex customization that includes object data, data service provider, logic service provider, and buyer experience extensions.
- For more information about writing a logic service provider extension and configuring your storefront to reference your extension, see [Extend or Override a Default Logic Service Provider](#).

### Example: **Example: Fetch a Custom Field from Stored Payment Records**

Let's say that you extended the default `ccrz__E_StoredPayment__c` object to add a custom field, `PurchaseOrderApprover__c`. To include this field's data in the `ccrz.ccApiStoredPayment.fetch` return data, extend the default `ccrz.ccServiceStoredPayment` data service provider class.

```
global class exampleServiceStoredPaymentExtension extends ccrz.ccServiceStoredPayment
{
    global override Map<String, Object> getFieldsMap(Map<String, Object> inputData) {
        ccrz.ccLog.log(LoggingLevel.DEBUG, 'E:exampleServiceStoredPaymentExtension,
'getFieldsMap');
        // Include all the default return data fields
        inputData = super.getFieldsMap(inputData);
        String objectFields = (String)inputData.get(ccrz.ccService.OBJECTFIELDS);
        // Add the custom field PurchaseOrderApprover__c
        inputData.put(ccrz.ccService.OBJECTFIELDS, objectFields +=
```


```

',PurchaseOrderApprover__c');
    ccrz.ccLog.log(LoggingLevel.DEBUG, 'E:exampleServiceStoredPaymentExtension,
'getFieldsMap');
    return super.handleFieldSizes(inputData);
}
}

```

### Example: Example: Prevent Deletion of a Stored Payment with an Active Subscription

Return an exception when a buyer tries to delete a stored payment that's used with an active subscription. This extension overrides the default `ccrz.ccLogicStoredPaymentRemove.deleteStoredPayment` logic service provider method to evaluate the `ccrz__E_Subscription__c` records that the `ccrz.ccLogicStoredPaymentRemove.fetchStoredPayment` method returned.

 **Note:** This example is compatible with only B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12). In previous versions of the managed package, the `ccrz.ccLogicStoredPaymentRemove` class doesn't return subscription data.

```

global class exampleLogicStoredPaymentRemoveExtension extends
ccrz.ccLogicStoredPaymentRemove {
    // Exception class that represents the exception case for a related active
    subscription
    global class RelatedSubscriptionException extends Exception { }

    global virtual override Map<String, Object> deleteStoredPayment (Map<String, Object>
inputData) {
        ccrz.ccLog.log(LoggingLevel.DEBUG, 'M:E',
'exampleLogicStoredPaymentRemoveExtension.deleteStoredPayment');

        // Check whether the subscription list is Map<String, Object> or sObject
        Boolean isSubMSO = !ccrz.ccUtil.isTrue(ccrz.ccApi.lookUpSz(inputData,
ccrz.ccAPISubscription.ENTITYNAME, ccrz.ccApi.SZ_SKIPTRZ));

        // Get the list of subscriptions
        List<Object> relatedSubscriptions = (List<Object>)
inputData.get(ccrz.ccAPISubscription.SUBSCRIPTIONS);

        if (ccUtil.isEmpty(relatedSubscriptions)) {
            for(Object s : relatedSubscriptions) {
                String subStatus = null;
                String subStoredPayment = null;
                if (isSubMSO) { // handle skiptrz = false
                    subStatus = (String)((Map<String,
Object>)s).get('subscriptionStatus');
                    subStoredPayment = (String)((Map<String,
Object>)s).get('storedPayment');
                } else { // handle skiptrz = true
                    subStatus = ((ccrz__E_Subscription__c)s).ccrz__SubscriptionStatus__c;

                    subStoredPayment =
((ccrz__E_Subscription__c)s).ccrz__StoredPayment__c;
                }
                // Start looking for active subscriptions
                if ((ccrz.cc_hk_Subscriptions.PARAM_STATUS_ACTIVE == subStatus) &&

```

```

(null != subStoredPayment)) {
    // Throw an exception when an active subscription is found
    ccrz.ccLog.log(LoggingLevel.DEBUG, 'ERR', 'Active related
subscription found.');
```

```

        throw new RelatedSubscriptionException('Active related subscription
found.');
```

```

    }
}

if(ccrz.ccUtil.isKeyValued(inputData,
ccrz.ccAPIStoredPayment.STOREDPAIDMENTSLLIST) && ccrz.ccUtil.isNotEmpty((List<object>)
inputData.get(ccrz.ccAPIStoredPayment.STOREDPAIDMENTSLLIST))){
    List<ccrz__E_StoredPayment__c>
stps=(List<ccrz__E_StoredPayment__c>) inputData.remove(ccrz.ccAPIStoredPayment.STOREDPAIDMENTSLLIST);

    delete stps;

    inputData.put(ccrz.ccAPI.SUCCESS, true);
}

ccrz.ccLog.log(LoggingLevel.DEBUG, 'M:X',
'exampleLogicStoredPaymentRemoveExtension.deleteStoredPayment');
return inputData;
}
}

```

## ccrz.ccApiStorefront

Query storefront association records, which link between a particular account, a particular storefront, and a particular Experience Cloud site.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccApiStorefront](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz\\_\\_E\\_StorefrontAssociation\\_\\_c](#)

[Create Associations between Multiple Storefronts and Experience Cloud Sites](#)

## ccrz.ccApiSubProdTerm

Query subscription options that are available for a specific product.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccApiSubProdTerm](#)

**Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[Offer Subscriptions for Products on Your Storefront](#)

## ccrz.ccApiSubscPage

Query subscriber page records, and handle the case where a specified subscriber page isn't found.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

IN THIS SECTION:

[fetch](#)

Executes a SOQL query of subscriber page records, and returns a list of records that match you query.

[notFound](#)

Returns a reference to the `ccrz.CCFileNotFound` Visualforce page that redirects a buyer when a specified subscriber page isn't found. Invoke this method when the `ccrz.ccApiSubscPage.fetch` method doesn't return any results.

SEE ALSO:

[Salesforce Help Create a Subscriber Page](#)

### ccrz.ccApiSubscPage.fetch


Executes a SOQL query of subscriber page records, and returns a list of records that match you query.

### Compatibility

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceSubscPage`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#).

 **Note:** Specific data size requests don't apply to this method. The same fields are always returned for each object, regardless of any specific data size request in your input map.


### `ccrz.ccApiSubscPage.PAGEID`

String that specifies the Salesforce ID of a subscriber page record to query.

 **Note:** This key takes precedence over `ccrz.ccApiSubscPage.PAGEIDLIST` and any other inputs.

### `ccrz.ccApiSubscPage.PAGEIDLIST`

`Set<String>` of Salesforce IDs of subscriber page records to query.

 **Note:** This key takes precedence over any other inputs except `ccrz.ccApiSubscPage.PAGEID`.

**ccrz.ccApiSubscPage.PAGEKEY**

String that specifies the key of a subscriber page to query. This value corresponds to the `ccrz__PageKey__c` field on a `ccrz__E_Subscriber_Page__c` record.

**ccrz.ccApiSubscPage.STOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiSubscPage.THEDATE**

Date for filtering the query by the `ccrz__StartDate__c` and `ccrz__EndDate__c` range on `ccrz__E_Subscriber_Page__c` records.

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiSubscPage.PAGELIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_Subscriber_Page__c` record.

 **Note:** Each subscriber page record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.

 **Example: Query by Page ID**

Fetch a subscriber page by its Salesforce ID.

```
Map<String, Object> subscriberPageQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiSubscPage.PAGEID => 'subscriber_page_ID'
};

try {
    Map<String, Object> subscriberPagesFetched =
ccrz.ccApiSubscPage.fetch(subscriberPageQuery);
```

```

    if (subscriberPagesFetched.get(ccrz.ccApiSubscPage.PAGELIST) != null) {
        List<Map<String, Object>> outputPagesList = (List<Map<String, Object>>)
subscriberPagesFetched.get(ccrz.ccApiSubscPage.PAGELIST);
        // Parse the returned pages data
    }
} catch (Exception e) {
    // Error handling...
}

```

### Example: Example: Query by Page IDs

Fetch multiple subscriber pages by their Salesforce IDs.

```

Map<String, Object> subscriberPageQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiSubscPage.PAGEIDLIST => new Set<String> {'subscriber_page_ID_1',
'subscriber_page_ID_2'}
};

try {
    Map<String, Object> subscriberPagesFetched =
ccrz.ccApiSubscPage.fetch(subscriberPageQuery);
    if (subscriberPagesFetched.get(ccrz.ccApiSubscPage.PAGELIST) != null) {
        List<Map<String, Object>> outputPagesList = (List<Map<String, Object>>)
subscriberPagesFetched.get(ccrz.ccApiSubscPage.PAGELIST);
        // Parse the returned pages data
    }
} catch (Exception e) {
    // Error handling...
}

```

### Example: Example: Query by Page Key and Date

Fetch subscriber pages that match a specific page key value and date range.

```

Map<String, Object> subscriberPageQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiSubscPage.PAGEKEY => 'pageKey',
    ccrz.ccApiSubscPage.THEDATE => Date.today().addDays(-5)
};

try {
    Map<String, Object> subscriberPagesFetched =
ccrz.ccApiSubscPage.fetch(subscriberPageQuery);
    if (subscriberPagesFetched.get(ccrz.ccApiSubscPage.PAGELIST) != null) {
        List<Map<String, Object>> outputPagesList = (List<Map<String, Object>>)
subscriberPagesFetched.get(ccrz.ccApiSubscPage.PAGELIST);
        // Parse the returned pages data
    }
} catch (Exception e) {
}

```

```
// Error handling...
}
```

SEE ALSO:

[ccrz\\_\\_E\\_Subscriber\\_Page\\_\\_c](#)

[Salesforce Help Create a Subscriber Page](#)

## ccrz.ccApiSubscPage.notFound

Returns a reference to the `ccrz.CCFileNotFound` Visualforce page that redirects a buyer when a specified subscriber page isn't found. Invoke this method when the `ccrz.ccApiSubscPage.fetch` method doesn't return any results.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> notFound (Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

`ccrz.ccLogicSubscPageNotFound`

### Inputs

None

### Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiSubscPage.PAGELIST**


List<Map<String, Object>> that contains one Map<String, Object>, which represents the `ccrz__E_Subscriber_Page__c` record for the `ccrz.CCFileNotFound` page.

## ccrz.ccApiSubscription

Query subscription records associated with placed orders.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccApiSubscription](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[Offer Subscriptions for Products on Your Storefront](#)

## ccrz.ccApiTerms

Query definitions of terms and conditions shown to the buyer during checkout.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

IN THIS SECTION:

[fetch](#)

Executes a SOQL query of terms records, and returns a list of records that match you query.


### ccrz.ccApiTerms.fetch

Executes a SOQL query of terms records, and returns a list of records that match you query.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceTerms`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies [options for how the method returns data](#).

 **Note:** Specific data size requests don't apply to this method. The same fields are always returned for each object, regardless of any specific data size request in your input map.

**ccrz.ccApiTerms.LOCALE**

String that specifies one of your [storefront's supported locales](#).

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccApiTerms.STOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiTerms.TYPE**

String that specifies a type of term for filtering the query, such as *All*, *Billing*, or *Shipping*.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiTerms.TERMLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_Term__c` record.

 **Note:**

- Each term record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.
- The return data includes only one term record of each type of term.

 **Example: Example**

Fetch term records where:

- Type is Shipping
- Storefront is DefaultStore
- Locale is en\_US

For each returned term, capture its title in a string variable.

```
Map<String, Object> termsFetchQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiTerms.STOREFRONT => 'DefaultStore',
    ccrz.ccApiTerms.TYPE => 'Shipping',
    ccrz.ccApiTerms.LOCALE => 'en_US'
};

try {
    Map<String, Object> termsResults = ccrz.ccApiTerms.fetch(termsFetchQuery);
    if (termsResults.get(ccrz.ccApiTerms.TERMLIST) != null) {
        List<Map<String, Object>> outputTermsList = (List<Map<String, Object>>)
termsResults.get(ccrz.ccApiTerms.TERMLIST);
        for (Map<String, Object> term: outputTermsList) {
            String termTitle = (String)term.get('title');
        }
    }
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[Salesforce Help Set Up Terms and Conditions on the Checkout Page](#)

## ccrz.ccApiTransactionPayment

Query details of payments made when buyers complete checkout.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

#### [fetch](#)

Executes a SOQL query of transaction payment records, and returns a list of records that match you query.


### `ccrz.ccApiTransactionPayment.fetch`

Executes a SOQL query of transaction payment records, and returns a list of records that match you query.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> fetch (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

```
ccrz.ccServiceTransactionPayment
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.


 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#).

 **Note:** Specific data size requests don't apply to this method. The same fields are always returned for each object, regardless of any specific data size request in your input map.

### `ccrz.ccApiTransactionPayment.BYASC`

Boolean

Value	Usage
<code>true</code>	Sort the returned transaction payment records by <code>ccrz.ccApiTransactionPayment.BYNAME</code> or <code>ccrz.ccApiTransactionPayment.BYTIMESTAMP</code> criteria in ascending order.
<code>false</code> (default)	Sort the returned transaction payment records in descending order.

### `ccrz.ccApiTransactionPayment.BYNAME`

Boolean

Value	Usage
<code>true</code> (default)	Sort the returned transaction payment records by the Name field.
<code>false</code>	Sort the returned transaction payment records other criteria.

### `ccrz.ccApiTransactionPayment.BYTIMESTAMP`

Boolean

Value	Usage
<code>true</code>	Sort the returned transaction payment records by the <code>ccrz__TransactionTS__c</code> field.
<code>false</code> (default)	Sort the returned transaction payment records other criteria.

**ccrz.ccApiTransactionPayment.FROMDATE**

Locale-specific string of a date value, such as `3/10/2015` for the `en_US` locale. The return data includes only the transaction payments where the `ccrz__TransactionTS__c` value occurs *after* the specified date. The `ccrz.ccServiceTransactionPayment` data service provider converts this string to a datetime.

**ccrz.ccApiTransactionPayment.ID**

String that specifies the Salesforce ID of the transaction payment record to query.

**ccrz.ccApiTransactionPayment.IDS**

`Set<String>` that specifies the Salesforce IDs of the transaction payment records to query.

**ccrz.ccApiTransactionPayment.INVOICESFOR**

`Set<String>` that specifies the Salesforce IDs of invoice records. The return data includes only the transaction payments that have applied for the specified invoices.

**ccrz.ccApiTransactionPayment.INVOICESTO**

`Set<String>` that specifies the Salesforce IDs of invoice records. The return data includes only the transaction payments that have applied to the specified invoices.

**ccrz.ccApiTransactionPayment.ORDERS**

`Set<String>` that specifies the Salesforce IDs of order records. The return data includes only the transaction payments associated with the specified orders.

**ccrz.ccApiTransactionPayment.STOREFRONT**

String that specifies the name of a storefront for filtering the query.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiTransactionPayment.SUBSCRIPTIONS**

`Set<String>` that specifies the Salesforce IDs of subscription records. The return data includes only the transaction payments associated with the specified subscriptions.

**ccrz.ccApiTransactionPayment.TODATE**

Locale-specific string of a date value, such as `3/10/2015` for the `en_US` locale. The return data includes only the transaction payments where the `ccrz__TransactionTS__c` value occurs *before* the specified date. The `ccrz.ccServiceTransactionPayment` data service provider converts this string to a datetime.

## Outputs


`Map<String, Object>` that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, consider rolling back the API transaction to a previous savepoint.</p>

**ccrz.ccApiTransactionPayment.TRANSACTIONPAYMENTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_TransactionPayment__c` record.



**Note:** Each transaction payment record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.

**Example: Example**

Fetch transaction payment records for a pair of specified order IDs, and capture the first result's method of payment.

```
Map<String, Object> transactionPaymentQuery = new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiTransactionPayment.ORDERS => new Set<String>{'order_ID_1', 'order_ID_2'}
};

try {
    Map<String, Object> transactionPaymentResults =
ccrz.ccApiTransactionPayment.fetch(transactionPaymentQuery);
    if
(transactionPaymentResults.get(ccrz.ccApiTransactionPayment.TRANSACTIONPAYMENTLIST)
!= null) {
        List<Map<String, Object>> outputPaymentsList = (List<Map<String, Object>>)
transactionPaymentResults.get(ccrz.ccApiTransactionPayment.TRANSACTIONPAYMENTLIST);
        String paymentTypes = (String) outputPaymentsList[0].get('accountType');
    }
} catch (Exception e) {
    // Error handling...
}
```

SEE ALSO:

[Salesforce Help Set Up Payment Types for Storefront Purchases](#)

[Salesforce Help Generate an Invoice for a Buyer's Order](#)

## ccrz.ccApiUser

Query, create, or modify user records that have B2B Commerce for Visualforce custom fields.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

### IN THIS SECTION:

#### [createUser](#)

Creates a user record from the fields and values that you specify. To insert the user after creating it, call the `registerUser` method, which invokes the `Site.createExternalUser` method.

#### [fetch](#)

Executes a SOQL query of user records and returns a list of users that match your query.

#### [getProfileData](#)

Returns user, contact, and account record data for populating fields on a buyer's My Account page.

#### [login](#)

Authenticates a user into a storefront.

#### [registerUser](#)

Creates the user, contact, and account records for a buyer who's registering on your storefront.

#### [revise](#)

Updates fields on an existing user record.

#### [updateProfileData](#)

Updates user or contact records with new values that a buyer specifies on the My Account page.

### SEE ALSO:

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers Site Class](#)


## ccrz.ccApiUser.createUser

Creates a user record from the fields and values that you specify. To insert the user after creating it, call the `registerUser` method, which invokes the `Site.createExternalUser` method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> createUser (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider


`ccrz.ccLogicUserCreateUser`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`


The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.




 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiUser.USER_NEW`

`Map<String, Object>` that describes fields and values for creating the [user record](#). Only certain account fields are writable when creating the record.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

Field	Type	Corresponding User Object Field Name
companyName	String	Company
companyType	String	<code>ccrz__CompanyType__c</code>
contactTypeRole	String	<code>ccrz__ContactTypeRole__c</code>
currencyCode	String	<code>ccrz__CC_CurrencyCode__c</code> and <code>DefaultCurrencyIsoCode</code>  <b>Note:</b> <code>DefaultCurrencyIsoCode</code> is set only when the org is configured for multiple currencies.
department	String	Department
firstName	String	FirstName
industry	String	<code>ccrz__User_Industry__c</code>

Field	Type	Corresponding User Object Field Name
language	String	LocaleSidKey and LanguageLocaleKey   <b>Note:</b> The <code>ccrz__Util__CustomSettingsLanguageLocaleKey</code> method filters the specified LanguageLocaleKey.
lastName	String	LastName
primaryPhone	String	Phone
salutation	String	ccrz__User_Salutation__c
timeZone	String	TimeZoneSIDKey   <b>Note:</b> This value is set only when the <a href="#">Show Time Zone configuration of the Registration module is TRUE</a> in your storefront configuration settings.
title	String	Title
username	String	SenderEmail and Nickname   <b>Note:</b> By default, the Experience Cloud site nickname matches the specified username, plus a timestamp to make the value unique.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiUser.USER\_NEW**

User sObject, created from the specified field values.


**ccrz.ccApiUser.fetch**

Executes a SOQL query of user records and returns a list of users that match your query.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

**Signature**

```
global static Map<String, Object> fetch(Map<String, Object>)
```

**Service Layer Classes****Data Service Provider**

ccrz.ccServiceUser

**Inputs (Required)**

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.


**Inputs (Optional)**

The input map can also include the following keys:



**ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies [options for how the method returns data](#).

 **Note:** Specific data size requests don't apply to this method. The same fields are always returned for each object, regardless of any specific data size request in your input map.

**ccrz.ccApiUser.ID\_LIST**

Set<String> of specific user IDs to query.

```
new Set<String>{'User_ID_1', 'User_ID_2'}
```

**ccrz.ccApiUser.ID**

String of a specific user ID to query.

**ccrz.ccApiUser.USER\_EMAIL**

String that specifies an email address whose user record you want to query.

**ccrz.ccApiUser.USER\_NAME**

String that specifies a username whose user record you want to query.

## Outputs


Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiUser.USERLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [user record](#).

 **Note:** Each user record in the return data includes the same set of fields, regardless of any specific `ccrz.ccApi.SIZING` input data size request.

### Example: Example

Fetch the user record for a specified user ID.

```
Map<String, Object> userFetchQuery = new Map<String, Object> {
    ccrz.ccApi.API_VERSION => ccrz.ccApi.CURRENT_VERSION,
    ccrz.ccApiUser.ID => 'User_ID'
};
```

```

Map<String, Object> userFetchData = ccrz.ccApiUser.fetch(userFetchQuery);

System.debug('userFetchData KEYS=' + JSON.serializePretty(userFetchData.keySet()));
Boolean success = (Boolean)userFetchData.get(ccrz.ccApi.SUCCESS);

if(success) {
    List<Map<String, Object>> users = (List<Map<String,
Object>>)userFetchData.get(ccrz.ccApiUser.USERLIST);
    System.debug('users=' + JSON.serializePretty(users));
} else {
    List<ccrz.cc_bean_Message> messages =
(List<ccrz.cc_bean_Message>)userFetchData.get(ccrz.ccApi.MESSAGES);
    System.debug('messages=' + messages);
}

```


## ccrz.ccApiUser.getProfileData

Returns user, contact, and account record data for populating fields on a buyer's My Account page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> getProfileData(Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider

```
ccrz.ccLogicUserGetProfileData
```

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

**ccrz.ccApiUser.USER\_ID**

String of a specific user ID whose profile data you want to return.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

## Outputs

Map<String, Object> that can include the following keys:

**ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

**ccrz.ccApi.MESSAGES**

List<[ccrz.cc\\_bean\\_Message](#)>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiUser.USER\_DATA\_MAP**

Map<String, Object> of [user record data](#).

**ccrz.ccApiUser.USER\_PROFILE\_ACCOUNT**

Map<String, Object> of [account record data](#).

**ccrz.ccApiUser.USER\_PROFILE\_CONTACT**

Map<String, Object> of [contact record data](#).


## ccrz.ccApiUser.login

Authenticates a user into a storefront.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

### Signature

```
global static Map<String, Object> login(Map<String, Object>)
```

### Service Layer Classes

#### Logic Service Provider


`ccrz.ccLogicUserLogin`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### `ccrz.ccApiUser.USER_NAME`

String that specifies the username for logging in.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

#### `ccrz.ccApiUser.USER_PWD`

String that specifies the user's password for logging in.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

## Inputs (Optional)

The input map can also include the following keys:

### **ccrz.ccApiUser.CART\_ENCID**

String that specifies an encrypted ID of a cart to transfer to the user's session.

### **ccrz.ccApiUser.LOGIN\_START\_URL**

String that specifies a storefront URL to immediately navigate the buyer to after logging in. This URL value can include a requested Visualforce page and [any parameters you want to set](#). If you don't specify this key, a buyer is sent to the Home page by default.



#### Note:

- B2B Commerce for Visualforce doesn't support immediately navigating a buyer to the Checkout page. If you try to immediately navigate to the Checkout page, the URL resolves to the Shopping Cart page.
- This method validates the requested URL for acceptable parameters, and returns the final redirect URL in the `ccrz.ccApiUser.LOGIN_PAGE` return data key.

### **ccrz.ccApiUser.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

Map<**String**, **Object**> that can include the following keys:

### **ccrz.ccApi.API\_VERSION**


Integer that indicates which API version was used for the query.

### **ccrz.ccApi.MESSAGES**

List<[ccrz.cc\\_bean\\_Message](#)>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### **ccrz.ccApiUser.LOGIN\_PAGE**

String that specifies the storefront page URL to navigate the buyer to after logging in. This value can include necessary parameters that were omitted from the `ccrz.ccApiUser.LOGIN_START_URL` input key.


## ccrz.ccApiUser.registerUser

Creates the user, contact, and account records for a buyer who's registering on your storefront.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

## Signature

```
global static Map<String, Object> registerUser (Map<String, Object>)
```

## Service Layer Classes

### Logic Service Provider

`ccrz.ccLogicUserRegister`

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiUser.USER_REGISTER_JSON`

JSON-formatted string of fields and values for the new user that's registering.

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApi.USER\_ID**

String that specifies the ID of the newly created user record.


**ccrz.ccApiUser.revise**

Updates fields on an existing user record.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

 **Warning:** This method doesn't allow a guest user to update records. If you're creating subscriber code that requires a guest user to update records, you have the following options:

- If you're using B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11), invoke the [ccrz.ccLogic.updateObjects](#) utility method.
- Execute the DML operation in your custom Apex class using the [without sharing](#) accessor.

**Signature**

```
global static Map<String, Object> revise (Map<String, Object>)
```

## Service Layer Classes

### Data Service Provider

`ccrz.ccServiceUser`

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### Inputs (Optional)

The input map can also include the following keys:

#### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiUser.fetch` for returning the updated user record.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiUser.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

#### `ccrz.ccApiUser.USERLIST`

`List<Map<String, Object>>`, where the first `Map<String, Object>` represents a user record whose field values you want to update.



**Note:** This method process only the first `Map<String, Object>` in the list, and ignores any others. Follow the same format for user record data that the `ccrz.ccApiUser.fetch` method returns.

## Outputs

`Map<String, Object>` that can include the following keys:

#### `ccrz.ccApi.API_VERSION`


Integer that indicates which API version was used for the query.

#### `ccrz.ccApi.SUCCESS`

Boolean


Value	Usage
true	The call completed.



Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApi.USERLIST**

List<Map<String, Object>>, where the first Map<String, Object> represents the updated [user record](#).

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiUser.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```


**ccrz.ccApiUser.updateProfileData**

Updates user or contact records with new values that a buyer specifies on the My Account page.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** You can still [call older versions of the API for this method](#), which can accept different input keys or return different output keys. Any differences in behavior for older versions aren't documented in this topic.

**Signature**

```
global static Map<String, Object> updateProfileData (Map<String, Object>)
```

**Service Layer Classes****Logic Service Provider**

[ccrz.ccLogicUserUpdateProfileData](#)

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApiUser.USER_DATA_MAP`

`Map<String, Object>` of the user record data to update.

### `ccrz.ccApiUser.USER_PROFILE_CONTACT`

`Map<String, Object>` of the user's contact record data to update.

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.API_VERSION`

Integer that indicates which API version was used for the query.

### `ccrz.ccApi.SUCCESS`

Boolean


Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

## ccrz.ccApiWishList

Query `ccrz__E_Cart__c` records where the `ccrz__CartType__c` field is `WishList`.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccApiWishList](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)


## Data Service Provider Classes

---

B2B Commerce for Visualforce provides several default *data service providers*: Apex classes that query and transform specific data for global API methods. For example, the `ccrz.ccServiceCart` class defines how B2B Commerce queries and transforms cart data for a `ccrz.ccApiCart.fetch` method call. To modify how a default class behaves, you can write subscriber code that extends or overrides the default class.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [Global API Data Service Layer](#)


 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## Logic Service Provider Classes

---

B2B Commerce for Visualforce provides several default *logic service providers*: Apex classes that describe the business logic for how specific global API methods behave. For example, the `ccrz.ccLogicCartAddTo` class defines how B2B Commerce adds products to a cart from a `ccrz.ccApiCart.addTo` method call. To modify the default logic for your organization, you can write subscriber code that extends or overrides the default logic.

Logic service provider classes differ from global API classes in a few important ways:

- Logic service provider definitions include the `virtual` definition modifier, which means that subscriber code can extend or override them.
- All methods in a logic service provider class execute in sequence. Each method accepts a `Map<String, Object>` and passes that map to the next method in the chain. This map is known as the *working map*. The next method in the chain can add or remove fields in the working map.
  -  **Warning:** When you extend or override a logic service provider, be careful to account for the status of the working map in each method. Which fields does each method add? Which fields does each method return? If you omit or mishandle a field at any point in the method chain, your subscriber code can function differently than you expect.
- Sometimes, B2B Commerce for Visualforce updates logic service provider classes in a new version of the B2B Commerce managed package. In subscriber code, you generally extend or override a version-agnostic logic service provider class or method. The version declared in the [input map for the associated global API method](#) determines which version of the logic service provider executes.

## IN THIS SECTION:

[Platform Cache for Logic Service Provider Classes](#)

Some logic service provider classes invoke methods that generate a key or prepare data for platform cache. These methods help improve performance when the logic service providers request cached data.

[Extend or Override a Default Logic Service Provider](#)

As part of your organization's customization of default B2B Commerce for Visualforce functionality, you can extend or override a default logic service provider class and its methods.

[ccrz.ccLogic](#)

Defines the base methods that are available for all logic service provider classes.

[ccrz.ccLogicAccountCreateAccount](#)

Define the business logic for creating an account record from the fields and values that you specify.

[ccrz.ccLogicAccountGetAnonymous](#)

Define the business logic for returning the account associated with the storefront guest user.

[ccrz.ccLogicAccountGetCurrent](#)

Define the business logic for returning the account associated with the user currently logged in to the storefront.

[ccrz.ccLogicAccountValidateNew](#)

Define the business logic for checking new account records for missing fields.

[ccrz.ccLogicAddressCreateAddress](#)

Define the business logic for creating a contact address record from the fields and values that you specify. On the storefront, this logic service provider is invoked by default when a buyer enters new address information during line-level independence (LLI) checkout flow.

[ccrz.ccLogicAddressBookCreateAddressBook](#)

Define the business logic for creating an account address book record from the fields and values that you specify.

[ccrz.ccLogicAddressBookRemoveAddressBook](#)

Define the business logic for deleting account address book records.

[ccrz.ccLogicAttrFetch](#)

Define the business logic for filtering `ccrz__E_Attribute__c` records returned from the `ccrz.ccApiAttribute.fetch` method's SOQL query. This class also defines whether `ccrz__E_Attribute__c` return data is keyed by `ccrz__AttributeID__c` or Salesforce ID.

[ccrz.ccLogicCartAddTo](#)

Define the business logic for adding products to a cart.

[ccrz.ccLogicCartClone](#)

Define the business logic for creating a copy of a cart or wish list.

[ccrz.ccLogicCartCreate](#)

Define the business logic for creating a cart record. By default, the methods of this class create only one cart record at a time. Even though the methods accept and return a list of cart records, the methods evaluate only the first element in the list.

[ccrz.ccLogicCartCreateCartItemGroup](#)

Define the business logic for creating a shipping group of cart items for line-level independence (LLI) checkout.

[ccrz.ccLogicCartCreateWL](#)

Define the business logic for creating a wish list and wish list items based on a specific cart ID and its cart items.

[ccrz.ccLogicCartInitItemGroups](#)

Define the business logic for initializing shipping groups on a specific cart for line-level independence (LLI) checkout.

[ccrz.ccLogicCartGetActive](#)

Define the business logic for resolving which cart is active for the current user, account (including effective accounts), storefront, and currency.

[ccrz.ccLogicCartGetDeliveryDates](#)

Define the business logic for returning the available delivery dates for one or more shipping addresses.

[ccrz.ccLogicCartGetShippingOptions](#)

Define the business logic for returning the available shipping options for one or more shipping addresses.

[ccrz.ccLogicCartPlace](#)

Define the business logic for converting a cart to an order.

[ccrz.ccLogicCartPlaceAsync](#)

Define the business logic for converting a cart to an order using asynchronous Apex jobs.

[ccrz.ccLogicCartPrice](#)

Define the business logic for updating the price information for the items in a specified cart.

[ccrz.ccLogicCartPriceAsync](#)

Define the business logic for converting a cart to an order using asynchronous Apex jobs.

[ccrz.ccLogicCartProcessItemGroups](#)

Define the business logic for moving cart items between shipping groups for line-level independence (LLI) checkout.

[ccrz.ccLogicCartRemove](#)

Define the business logic for removing cart items, including a coupon, from a specified cart.

[ccrz.ccLogicCartRemoveCart](#)

Define the business logic for deleting carts, including wish lists.

[ccrz.ccLogicCartRemoveItemGroups](#)

Define the business logic for deleting shipping groups of cart items for line-level independence (LLI) checkout.

[ccrz.ccLogicCartRepair](#)

Define the business logic for reopening a cart where the status remains Repricing even after a repricing job completes. A cart can remain in this status when a repricing job is unsuccessful or no longer available in the org.

[ccrz.ccLogicCartSave](#)

Define the business logic for saving a cart, including invoking validation logic and updating cart fields for a selected effective account.

[ccrz.ccLogicCartSetActive](#)

Define the business logic for setting a single cart as the active cart for a particular owner, and updating all the owner's other open carts to inactive.

[ccrz.ccLogicCartTransfer](#)

Define the business logic for transferring ownership of a cart to another user.

[ccrz.ccLogicCartValidate](#)

Define the business logic for determining whether a buyer is allowed to check out from the Shopping Cart page. This class also replaces the legacy `ccrz.cc_api_CartExtension.allowCheckout` and `ccrz.cc_api_CartExtension.getCartMessages` methods.

[ccrz.ccLogicCategoryGetTree](#)

Define the business logic for returning a generated category tree from a public cache.

[ccrz.ccLogicCategoryUrlFor](#)

Define the business logic for returning the friendly URLs for categories.

[ccrz.ccLogicCouponApply](#)

Define the business logic for adding and applying a coupon to a specific shopping cart.

[ccrz.ccLogicCouponRemove](#)

Define the business logic for removing coupons that are already applied to a specified shopping cart.

[ccrz.ccLogicCouponUpdateUse](#)

Define the business logic for removing coupons that are already applied to a specified shopping cart.

[ccrz.ccLogic18NGetAvailableLocales](#)

Define the business logic for returning a storefront's enabled locales.

[ccrz.ccLogic18NGetBrowserLocales](#)

Define the business logic for determining which locales the buyer's browser supports.

[ccrz.ccLogic18NGetLocale](#)

Define the business logic for returning the current buyer's locale.

[ccrz.ccLogic18NSetLocale](#)

Define the business logic for setting a new locale for the current authenticated buyer.

[ccrz.ccLogicOrderAuthCancel](#)

Define the business logic for determining whether a buyer can cancel an order.

[ccrz.ccLogicOrderCancel](#)

Define the business logic for changing the status of an order to `Cancelled`.

[ccrz.ccLogicOrderHistory](#)

Define the business logic for searching an account's order history, and fetching order records that match the search terms. This class also replaces the legacy `ccrz.cc_hk_Order.fetchOrderHistory` method.

[ccrz.ccLogicOrderGetOrdersToSplit](#)

Define the business logic for determining which orders to include in the `ccrz.ccSplitOrderJob` batch processing class.

[ccrz.ccLogicOrderReOrder](#)

Define the business logic for copying a completed order and its order items to a new cart and cart items. This class also replaces the legacy `ccrz.cc_hk_Order.reorder` method.

[ccrz.ccLogicOrderRestore](#)

Define the business logic for reopening a cart after the corresponding order fails when attempting to place the order asynchronously. This cart's status resets to `Open` and the cart becomes active. This logic service provider can only restore an order with a status of `Error` and that hasn't been restored previously.

[ccrz.ccLogicOrderSplitOrder](#)

Define the business logic for splitting orders by seller. To split orders by other criteria, extend this logic service provider class.

[ccrz.ccLogicPICreate](#)

Define the business logic for creating product index records.

[ccrz.ccLogicPIFetchEntitled](#)

Define the business logic for returning a buyer's entitled product IDs that correspond to product index records queried from a search string or specified categories.

[ccrz.ccLogicPIRemove](#)

Define the business logic for deleting product index records.

[ccrz.ccLogicPLEntLists](#)

Define the business logic for returning an account's entitled price lists.

[ccrz.ccLogicProductFind](#)

Define the business logic for returning and pricing entitled products that match a search string or that belong to a set of specified category IDs.

[ccrz.ccLogicProductFindFilters](#)

Define the business logic for returning filters and spec values for entitled products that match a search string or that belong to a set of specified category IDs.

[ccrz.ccLogicProductPricing](#)

Define the business logic for pricing products.

[ccrz.ccLogicProductUrlFor](#)

Define the business logic for returning the friendly URLs for products.

[ccrz.ccLogicSICreate](#)

Define the business logic for creating site index records.

[ccrz.ccLogicSIMapUrl](#)

Define the business logic for returning the category or product default URL that corresponds to a specified friendly URL.

[ccrz.ccLogicSIRemove](#)

Define the business logic for deleting site index records.

[ccrz.ccLogicStoredPaymentCreate](#)

Define the business logic for creating a stored payment record.

[ccrz.ccLogicStoredPaymentRemove](#)

Define the business logic for deleting stored payment records.

[ccrz.ccLogicUserCreateUser](#)

Define the business logic for creating a user record from the fields and values that you specify. This class doesn't insert the user. Instead, the `ccrz.ccLogicUserRegister` logic service provider class invokes the `Site.createExternalUser` method.

[ccrz.ccLogicUserGetProfileData](#)

Define the business logic for returning user, contact, and account record data for populating fields on a buyer's My Account page.

[ccrz.ccLogicUserLogin](#)

Define the business logic for authenticating a user into a storefront.

[ccrz.ccLogicUserRegister](#)

Define the business logic for creating the user, contact, and account records for a buyer who's registering on your storefront.

[ccrz.ccLogicUserUpdateProfileData](#)

Define the business logic for updating user or contact records with new values that a buyer specifies on the My Account page.

[ccrz.ccLogicWLAddTo](#)

Define the business logic for adding products to a wish list.

[ccrz.ccLogicWLCreate](#)

Define the business logic for creating a wish list.

[ccrz.ccLogicWLCreateCart](#)

Define the business logic for creating a cart record from an existing wish list.

[ccrz.ccLogicWLGetActive](#)

Define the business logic for resolving which wish list is active for the current user, account (including effective accounts), storefront, and currency.

[ccrz.ccLogicWLRemove](#)

Define the business logic for deleting a wish list.

[ccrz.ccLogicWLRemoveFrom](#)

Define the business logic for deleting wish list items from a specified wish list.

[ccrz.ccLogicWLSetActive](#)

Define the business logic for setting a single wish list as the active wish list for a particular owner, and updating all the owner's other wish lists to inactive.

SEE ALSO:

[Apex Class Definition](#)

## Platform Cache for Logic Service Provider Classes

Some logic service provider classes invoke methods that generate a key or prepare data for platform cache. These methods help improve performance when the logic service providers request cached data.

The following methods of the `ccrz.ccLogic` base class interact with cached data:

- `ccrz.ccLogic.prepareCacheData`
- `ccrz.ccLogic.prepareCacheKey`

The specific behavior for invoking these methods varies by logic service provider.

## Extend or Override a Default Logic Service Provider

As part of your organization's customization of default B2B Commerce for Visualforce functionality, you can extend or override a default logic service provider class and its methods.

For example, let's say that you want to customize the default logic for returning price lists that a particular account is entitled to.

1. In your org, create an Apex class for your customization.

```
global with sharing class customEntitledPriceListLogic extends ccrz.ccLogicPLEntLists
{ }
```



Tip:

- We recommend that you define and adopt a consistent naming convention for your organization's subscriber code. Include a common prefix, your project's name, and the name of the default class, such as `cc_projectName_defaultClassName`.
  - Specify an appropriate [sharing keyword](#), depending on the availability and permissions that you want the class to have.
  - Use the `extends` keyword since you're extending a default logic service provider class.
2. Inside the class definition, override the specific method of the `ccrz.ccLogicPLEntLists` logic provider class whose behavior you want to customize.

Let's say that you want to override the `fetchAccountGroupPriceListIDs` method, which returns the IDs of the price lists associated with an account group.

```
global with sharing class customEntitledPriceListLogic extends ccrz.ccLogicPLEntLists
{
    global virtual override Map<String, Object> fetchAccountGroupPriceListIDs (Map<String,
```



```
Object> inputData) { }
}
```

Because each logic service provider method accepts a `Map<String, Object>`, your method override must also accept a `Map<String, Object>` for input data.

3. Inside your method override, add your custom business logic. Make sure that you include a return statement at the end of your method override.

```
global with sharing class customEntitledPriceListLogic extends ccrz.ccLogicPLEntLists
{
    global virtual override Map<String, Object> fetchAccountGroupPriceListIDs (Map<String,
Object> inputData) {
        // Your custom logic
        return inputData;
    }
}
```



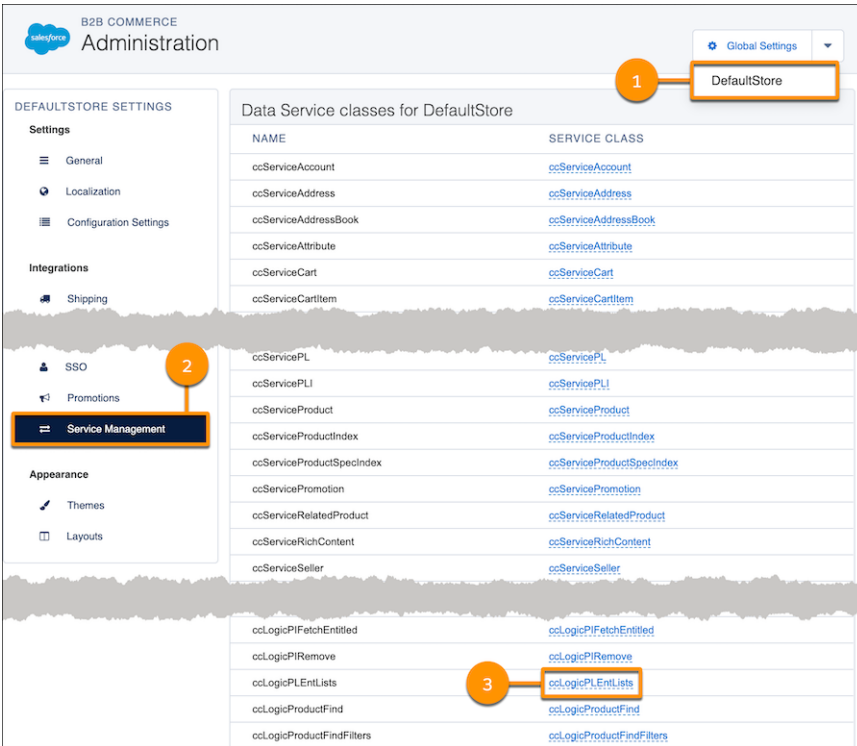
#### Tip:

- Since you're overriding a `virtual` method, you can [invoke the `super` keyword in your custom logic](#).
- To help you understand what data a method expects and returns by default, check out the reference for that particular method. During development, you can also serialize the default input and output data within your method override. This step can help you make sure that your extension code handles all expected inputs and outputs correctly so that your code executes successfully.

```
global with sharing class customEntitledPriceListLogic extends
ccrz.ccLogicPLEntLists {
    global virtual override Map<String, Object>
fetchAccountGroupPriceListIDs (Map<String, Object> inputData) {
        System.debug('-- inputData ' + JSON.serialize(inputData));
        Map<String, Object> returnData =
super.fetchAccountGroupPriceListIDs(inputData);
        System.debug('-- returnData ' + JSON.serialize(returnData));

        return returnData;
    }
}
```

4. Save your class, and resolve any compilation errors.
5. Update your storefront settings so that B2B Commerce for Visualforce references your override instead of the default logic service provider.



- a. On the CC Admin tab in your org, select your storefront (1).
- b. In the Storefront Settings menu, select **Service Management** (2).
- c. Scroll to the name of the logic service provider class that you extended, such as `ccLogicPLEntLists`, and click its link in the Service Class column (3).
- d. Enter the name of your custom class, prefixed with `c.`, such as `c.customEntitledPriceListLogic`, and click the checkmark  to save.

## ccrz.ccLogic

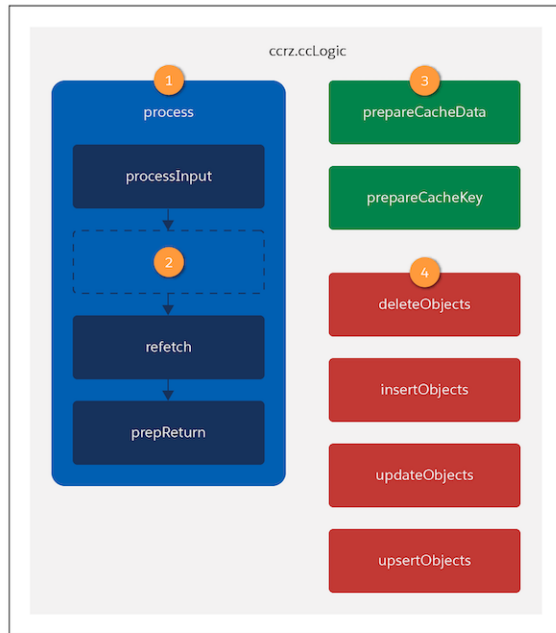
Defines the base methods that are available for all logic service provider classes.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

The methods of this class execute in the following order.



- The `process` method (1) invokes the chain of inner methods. Classes that extend the `ccrz.ccLogic` base class, including the default B2B Commerce for Visualforce logic service provider classes, typically add specific logic (2) as part of this process.
- Global utility methods for generating a key or preparing data for platform cache (3).
- In B2B Commerce for Visualforce Spring '20 (version 4.12), static utility methods (4) are available for performing DML operations on `sObjects`.

## Methods

### IN THIS SECTION:

#### [process](#)

Accepts the input map specified for a B2B Commerce for Visualforce global API method, and passes the map to the `ccrz.ccLogic.processInput` method. Specific logic service provider class methods that override this base method often call into other business logic from `process`.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [refetch](#)

Executes a follow-up fetch operation for data that was previously fetched, when appropriate. For example, if you're programmatically modifying cart data, maybe you want to return the updated cart records within the same API transaction.

#### [prepReturn](#)

Removes intermediate keys added to the working map by other logic methods, and returns the final data queried by the B2B Commerce for Visualforce global API method.

[prepareCacheData](#)

Evaluates and potentially modifies data before caching it.

[prepareCacheKey](#)

Generates a key for a particular type of cached data, such as category tree or page labels. This method returns a `Map<String, Object>` that contains fields that ensure that repeated calls always return the same set of data. The exact keys depend on the specific logic service provider that invokes this method.

[deleteObjects](#)

Deletes a specified `sObject` or list of `sObjects`.

[insertObjects](#)

Inserts a specified `sObject` or list of `sObjects`.

[updateObjects](#)

Updates a specified `sObject` or list of `sObjects`.

[upsertObjects](#)

Upserts a specified `sObject` or list of `sObjects`.

## ccrz.ccLogic.process

Accepts the input map specified for a B2B Commerce for Visualforce global API method, and passes the map to the `ccrz.ccLogic.processInput` method. Specific logic service provider class methods that override this base method often call into other business logic from `process`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs

This method accepts a `Map<String, Object>` of the input data for a B2B Commerce for Visualforce global API method. The specific keys included in the map depend on which global API method called the logic service provider.

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.



**Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogic.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs

This method accepts a `Map<String, Object>` of the input data for a B2B Commerce for Visualforce global API method. The specific keys included in the map depend on which global API method called the logic service provider.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

### **errorMessages**

`Map<String, Object>` that contains any keys that the B2B Commerce for Visualforce global API method requires but that weren't included in the original input data. For example, several global API methods require one or more record IDs in the input data.

If any required keys are missing, the method returns a `ccrz.ccApi.MissingInputException`.

## ccrz.ccLogic.refetch

Executes a follow-up fetch operation for data that was previously fetched, when appropriate. For example, if you're programmatically modifying cart data, maybe you want to return the updated cart records within the same API transaction.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

### Inputs

This method accepts a `Map<String, Object>`, which is a working map that contains the data copied from the original input map. The working map can also include the following keys:

#### **ccrz.ccApi.SZ\_REFETCH**

Boolean

Value	Usage
<i>true</i> (default)	<p>Check the value of the <code>ccrz.ccApi.SZ_REFETCH</code> key in the original input data, and execute the refetch operation if <code>ccrz.ccApi.SZ_REFETCH =&gt; true</code>. A useful function for checking this value is:</p> <pre>ccrz.ccApi.lookupSz (Map&lt;String, Object&gt; <b>inputData</b>, String <b>entityName</b>, String <b>refetchKey</b>)</pre> <p>For example, to check the value of the <code>ccrz.ccApi.SZ_REFETCH</code> key in an input map named <code>cartFetchQuery</code> for cart data:</p> <pre>ccrz.ccApi.lookupSz (Map&lt;String, Object&gt; <b>cartFetchQuery</b>, ccrz.ccApiCart.ENTITYNAME, ccrz.ccApi.SZ_REFETCH)</pre>
<i>false</i>	Ignore the value of the <code>ccrz.ccApi.SZ_REFETCH</code> key in the original input data.

## Outputs

This method returns a `Map<String, Object>` of the refetched data. The specific keys returned depend on which global API method is requesting the refetch operation.

SEE ALSO:

[Sizing and Scoping Options for a B2B Commerce for Visualforce API Method Call](#)

## ccrz.ccLogic.prepReturn

Removes intermediate keys added to the working map by other logic methods, and returns the final data queried by the B2B Commerce for Visualforce global API method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs

This method accepts a `Map<String, Object>`, which is a working map that contains the data copied from the original input map. The working map also contains any extra keys added by other methods in the logic chain.

## Outputs

This method returns a `Map<String, Object>`, which is the final data queried by the global API method.

## ccrz.ccLogic.prepareCacheData

Evaluates and potentially modifies data before caching it.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepareCacheData (Map<String, Object>)
```

## Inputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.CACHE_DATA`

`Map<String, Object>` that specifies data to cache for a particular object, such as the category tree or page labels.

### `ccrz.ccApi.CACHE_INP`

`Map<String, Object>` that contains input data for the cache query.

### `ccrz.ccApi.CACHE_KEY`

String that specifies a key for the cached data. The `ccrz.ccLogic.prepareCacheKey` method returns the value for this key.

### `ccrz.ccApi.CACHE_TTL`

Integer that specifies the number of seconds for the time to live (TTL) for a particular type of cached data, or how long that data remains in the cache. For example, a TTL value of `300` is 300 seconds, or five minutes. When the TTL expires, the data is no longer available in the cache.

## Outputs

`Map<String, Object>` that contains the input data.

### Example: Example

Override the default `ccrz.ccLogic.prepareCacheData` method so that:

- For a storefront named `storefront_name_1`, avoid caching data.
- For a storefront named `storefront_name_2`, set the TTL to half of its configured value.

```
global virtual override Map<String, Object> prepareCacheData (Map<String, Object>
.get(ccrz.ccApi.CACHE_TTL);
        if (null != customTTL) {
```



```

        Map<String, Object> cacheDataMap = new Map<String, Object>(inputData);
        cacheDataMap.put(ccrz.ccApi.CACHE_TTL, customTTL/2);
        return cacheDataMap;
    }
}
return inputData;
}

```

SEE ALSO:

[Salesforce Help Improve Storefront Performance with Platform Cache](#)

## ccrz.ccLogic.prepareCacheKey

Generates a key for a particular type of cached data, such as category tree or page labels. This method returns a `Map<String, Object>` that contains fields that ensure that repeated calls always return the same set of data. The exact keys depend on the specific logic service provider that invokes this method.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepareCacheKey (Map<String, Object>)
```

### Inputs

`Map<String, Object>` that contains all the keys passed to the specific logic service provider method that invokes this method. For example, when the `ccrz.ccLogicCategoryGetTree.getTree` method invokes this method, the input `Map<String, Object>` contains all the keys that `ccrz.ccLogicCategoryGetTree.getTree` requires, including `ccrz.ccApiCategory.CATEGORYLOCALE` and `ccrz.ccApiCategory.STOREFRONT`.

### Outputs

`Map<String, Object>` that contains the following key:

**ccrz.ccApi.CACHE\_KEYMAP**

Map<String, Object> that uniquely identifies the requested cache data and ensures that repeated calls always return the same set of data.

 **Note:** When this value is null or empty, the underlying data service provider class doesn't use platform cache.

 **Example: Example**

Override the default `ccrz.ccLogic.prepareCacheKey` method and add an entry to `ccrz.ccApi.CACHE_KEYMAP`. This entry makes the cache key unique for only the current day (GMT) so that requests to the same cache after midnight must use a new key.

```
global virtual override Map<String, Object> prepareCacheKey(Map<String, Object>


```

SEE ALSO:

[Salesforce Help Improve Storefront Performance with Platform Cache](#)


**ccrz.ccLogic.deleteObjects**

Deletes a specified sObject or list of sObjects.

**Compatibility**

This utility method is available *only* in the following versions of the managed package:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

 **Warning:** Be careful when invoking this method in subscriber code that extends or overrides the `ccrz.ccLogic` base class. This method executes destructive changes to your org's data.

**Signature**

```
global static void deleteObjects(sObject)
```

```
global static void deleteObjects(List<sObject>)
```

## Inputs

This method can accept either of the following parameters:

### **objectToDelete**

sObject to delete.

### **objectsToDelete**

List<sObject> to delete. Each sObject in the list must be the same type, such as `ccrz__E_Cart__c`.



**Note:** When executed in the context of a guest user, this method executes `without sharing` and can only modify records owned by `default guest user record owner`. The default guest user record owner is an Experience Cloud site user that you designate when you [resolve security alerts to keep your storefront secure by default](#).

## Outputs

This method doesn't return any data.

SEE ALSO:

[Salesforce Help Resolve Guest User Security Alerts to Keep Your Storefronts Secure by Default](#)

[Salesforce Help Create a Default Owner for Records Created by a Storefront Guest User](#)

## ccrz.ccLogic.insertObjects

Inserts a specified sObject or list of sObjects.

## Compatibility

This utility method is available *only* in the following versions of the managed package:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11



**Warning:** Be careful when invoking this method in subscriber code that extends or overrides the `ccrz.ccLogic` base class. This method executes destructive changes to your org's data.

## Signature

```
global static void insertObjects(sObject)
```

```
global static void insertObjects(List<sObject>)
```

## Inputs


This method can accept either of the following parameters:

### **objectToInsert**

sObject to insert.

**objectsToInsert**

`List<sObject>` to insert. Each `sObject` in the list must be the same type, such as `ccrz__E_Cart__c`.

-  **Note:** When executed in the context of a guest user, this method executes *without sharing* and can only modify records owned by [default guest user record owner](#). The default guest user record owner is an Experience Cloud site user that you designate when you [resolve security alerts to keep your storefront secure by default](#).

**Outputs**

This method doesn't return any data.

## SEE ALSO:

[Salesforce Help Resolve Guest User Security Alerts to Keep Your Storefronts Secure by Default](#)

[Salesforce Help Create a Default Owner for Records Created by a Storefront Guest User](#)


**ccrz.ccLogic.updateObjects**

Updates a specified `sObject` or list of `sObjects`.

**Compatibility**

This utility method is available *only* in the following versions of the managed package:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

-  **Warning:** Be careful when invoking this method in subscriber code that extends or overrides the `ccrz.ccLogic` base class. This method executes destructive changes to your org's data.

**Signature**

```
global static void updateObjects(sObject)
```

```
global static void updateObjects(List<sObject>)
```

**Inputs**


This method can accept either of the following parameters:

**objectToUpdate**

`sObject` to update.

**objectsToUpdate**

`List<sObject>` to update. Each `sObject` in the list must be the same type, such as `ccrz__E_Cart__c`.

 **Note:** When executed in the context of a guest user, this method executes `without sharing` and can only modify records owned by `default guest user record owner`. The default guest user record owner is an Experience Cloud site user that you designate when you [resolve security alerts to keep your storefront secure by default](#).

## Outputs

This method doesn't return any data.

SEE ALSO:

[Salesforce Help Resolve Guest User Security Alerts to Keep Your Storefronts Secure by Default](#)

[Salesforce Help Create a Default Owner for Records Created by a Storefront Guest User](#)


## ccrz.ccLogic.upsertObjects

Upserts a specified sObject or list of sObjects.

## Compatibility

This utility method is available *only* in the following versions of the managed package:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

 **Warning:** Be careful when invoking this method in subscriber code that extends or overrides the `ccrz.ccLogic` base class. This method executes destructive changes to your org's data.

## Signature

```
global static void upsertObjects(sObject)
```

```
global static void upsertObjects(List<sObject>)
```

## Inputs


This method can accept either of the following parameters:

### **objectToUpsert**

sObject to upsert.

### **objectsToUpsert**

List<sObject> to upsert. Each sObject in the list must be the same type, such as `ccrz__E_Cart__c`.

 **Note:** When executed in the context of a guest user, this method executes `without sharing` and can only modify records owned by `default guest user record owner`. The default guest user record owner is an Experience Cloud site user that you designate when you [resolve security alerts to keep your storefront secure by default](#).

## Outputs

This method doesn't return any data.

SEE ALSO:

[Salesforce Help Resolve Guest User Security Alerts to Keep Your Storefronts Secure by Default](#)

[Salesforce Help Create a Default Owner for Records Created by a Storefront Guest User](#)

## ccrz.ccLogicAccountCreateAccount

Define the business logic for creating an account record from the fields and values that you specify.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiAccount.createAccount](#)

## Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

[processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

[getOwnerAccount](#)

Calls the `ccrz.ccApiAccount.fetch` method for returning the storefront's portal account, which becomes the parent account for the new account. By default, the portal account is the owner of a new account when a buyer self-registers on your storefront.

[processAccount](#)

Constructs an account sObject from the input values.

[createAccount](#)

Executes the DML `insert` operation on the account record to create.

[refetch](#)

Call the `ccrz.ccApiAccount.fetch` method for fetching the created account.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicAccountCreateAccount.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiAccount.createAccount` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicAccountCreateAccount` class in the following order:

1. `processInput`
2. `getOwnerAccount`
3. `processAccount`
4. `createAccount`
5. `refetch`
6. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicAccountCreateAccount.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicAccountCreateAccount.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

**ccrz.ccApiAccount.ACCOUNT\_NEW**

`List<String>` that specifies fields and values for creating the [account record](#).

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

**ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicAccountCreateAccount.prepReturn method` removes this intermediate key from the working map.

SEE ALSO:

[Account \(B2B Commerce for Visualforce\)](#)



## ccrz.ccLogicAccountCreateAccount.getOwnerAccount

Calls the `ccrz.ccApiAccount.fetch` method for returning the storefront's portal account, which becomes the parent account for the new account. By default, the portal account is the owner of a new account when a buyer self-registers on your storefront.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> getOwnerAccount (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicAccountCreateAccount.processInput` method completes.

### Logic Performed

This method completes the following actions:

1. Passes the value of the `ccrz__Customer_Portal_Account_Name__c` field from `ccrz.cc_CallContext.storefrontSettings` as the `ccrz.ccApiAccount.NAME` input for `ccrz.ccApiAccount.fetch`.
2. Parses the `ccrz.ccApiAccount.fetch` output for the portal account and corresponding account group.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

#### **ccrz.ccLogicAccountCreateAccount.AG**

String that specifies the Salesforce ID of the account group for the storefront's portal account.

**ccrz.ccLogicAccountCreateAccount.OWNER\_ACCOUNT**

[Account](#) that represents the storefront's portal account.

SEE ALSO:

[ccrz.ccApiAccount.fetch](#)

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

[Salesforce Help Change Your Storefront's Portal Account Group](#)

**ccrz.ccLogicAccountCreateAccount.processAccount**

Constructs an account sObject from the input values.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> processAccount (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicAccountCreateAccount.getOwnerAccount` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiAccount.ACCOUNT\_NEW**

`List<String>` that specifies fields and values for creating the [account record](#). Only certain account fields are writable when creating the record:

Field	Required?	Type
billingAddress	Required	Map<String, Object> (see the following table)
companyName	Optional	String
customField1	Optional	String

Field	Required?	Type
customField2	Optional	String
firstName	Required	String
lastName	Required	String
primaryPhone	Required	String
shippingAddress	Required	Map<String, Object> (see the following table)
taxExemptFlag	Required	Boolean
userName	Required	String

The `billingAddress` and `shippingAddress` maps can contain the following writable fields:

Field	Required?	Type
address1	Required	String
address2	Optional	String
city	Required	String
country	Required	String
countryCode	Optional	String
postalCode	Required	String
state	Required	String
stateCode	Optional	String

#### **ccrz.ccLogicAccountCreateAccount.AG**

String that specifies the Salesforce ID of the account group for the storefront's portal account.

#### **ccrz.ccLogicAccountCreateAccount.OWNER\_ACCOUNT**

[Account](#) that represents the storefront's portal account.

## Logic Performed

This method sets the following fields on the account sObject based on the input data:

Account Field	Input Data
<code>ccrz__AccountGroup__c</code>	<code>ccrz.ccLogicAccountCreateAccount.AG</code>
BillingAddress	<code>billingAddress</code> field of <code>ccrz.ccApiAccount.ACCOUNT_NEW</code>

Account Field	Input Data
ccrz__CustomField1__c	customField1 field of ccrz.ccApiAccount.ACCOUNT_NEW
ccrz__CustomField2__c	customField2 field of ccrz.ccApiAccount.ACCOUNT_NEW
Description	userName field of ccrz.ccApiAccount.ACCOUNT_NEW
OwnerId	ccrz.ccLogicAccountCreateAccount.OWNER_ACCOUNT
Name	<ol style="list-style-type: none"> <li>1. If specified, companyName field of ccrz.ccApiAccount.ACCOUNT_NEW</li> <li>2. Otherwise, a concatenation of firstName and lastName fields of ccrz.ccApiAccount.ACCOUNT_NEW</li> </ol>
Phone	primaryPhone field of ccrz.ccApiAccount.ACCOUNT_NEW
ShippingAddress	shippingAddress field of ccrz.ccApiAccount.ACCOUNT_NEW
ccrz__TaxExemptAccount__c	taxExempt field of ccrz.ccApiAccount.ACCOUNT_NEW
ccrz__UsedByCloudcraze__c	true

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiAccount.ACCOUNT\_NEW**

[Account](#) that represents the constructed account sObject.

SEE ALSO:

[Account \(B2B Commerce for Visualforce\)](#)

## ccrz.ccLogicAccountCreateAccount.createAccount

Executes the DML `insert` operation on the account record to create.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createAccount (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicAccountCreateAccount.processAccount` method completes.


This method requires the following key in the working map:

**ccrz.ccApiAccount.ACCOUNT\_NEW**  
[Account](#) that represents the account sObject to insert.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApi.SUCCESS**  
 Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint.</a>

SEE ALSO:

[Inserting and Updating Records](#)

## ccrz.ccLogicAccountCreateAccount.refetch

Call the `ccrz.ccApiAccount.fetch` method for fetching the created account.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicAccountCreateAccount.createAccount` method completes.

This method requires the following keys in the working map:

### `ccrz.ccApi.SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAccount.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```



**Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

### `ccrz.ccApiAccount.ACCOUNT_NEW`

`Account` that represents the created account sObject.

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiAccount.fetch` outputs.

SEE ALSO:

[ccrz.ccApiAccount.fetch](#)

## `ccrz.ccLogicAccountCreateAccount.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicAccountCreateAccount.refetch` method completes.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApi.SIZING`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicAccountCreateAccount.AG`
- `ccrz.ccLogicAccountCreateAccount.OWNER_ACCOUNT`

## ccrz.ccLogicAccountGetAnonymous

Define the business logic for returning the account associated with the storefront guest user.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiAccount.getAnonymous](#)

## Methods

### IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method calls the `ccrz.ccApiAccount.fetch` method for returning the storefront's guest user account, and stores the account data in platform cache.

## ccrz.ccLogicAccountGetAnonymous.process

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method calls the `ccrz.ccApiAccount.fetch` method for returning the storefront's guest user account, and stores the account data in platform cache.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)


This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiAccount.getAnonymous` method.

This method requires the following key in the input data:



**ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

**Inputs (Optional)**

The input map can also include the following keys:

**ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

**Logic Performed**

This method completes the following actions:

1. Invokes the `ccrz.ccLogic.pullFromCache` public utility method, which calls `ccrz.ccLogic.prepareCacheKey`. This method references the value of the Anonymous Account Cache TTL configuration of the Platform Cache module (`cache.anonacctttl`) in your storefront configuration settings.
2. Passes input data to the `ccrz.ccApiAccount.fetch` method, where the `ccrz.ccApiAccount.NAME` input key references the value of the Anonymous Account Name global setting. By default, this anonymous account name is `CCAnonymous`.
3. Calls a public override of the `ccrz.ccLogic.prepareCacheData` method.
4. Calls the `ccrz.ccLogic.pushIntoCache` public utility method for inserting the cached anonymous account information.

**Outputs**

`Map<String, Object>` that includes the `ccrz.ccApiAccount.ACCOUNTS` output from `ccrz.ccApiAccount.fetch`.

SEE ALSO:

[ccrz.ccApiAccount.fetch](#)

[Salesforce Help: Allow a B2B Commerce for Visualforce Guest Buyer to Browse Your Storefront](#)

[Salesforce Help Improve Storefront Performance with Platform Cache](#)

**ccrz.ccLogicAccountGetCurrent**

Define the business logic for returning the account associated with the user currently logged in to the storefront.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiAccount.getCurrent](#)

## Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method calls the `ccrz.ccApiAccount.fetch` method for returning the current storefront user's account.

## ccrz.ccLogicAccountGetCurrent.process

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method calls the `ccrz.ccApiAccount.fetch` method for returning the current storefront user's account.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiAccount.getCurrent` method.

This method requires the following key in the input data:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

The input map can also include the following keys:

### `ccrz.ccApi.API_SIZING`

`Map<String, Object>` that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

## Logic Performed

This method completes the following actions:

1. If the current user is a guest, this method invokes `ccrz.ccApiAccount.getAnonymous` for returning the guest user's account information.
2. Otherwise, this method passes input data to the `ccrz.ccApiAccount.fetch` method, where the `ccrz.ccApiAccount.NAME` input key references `ccrz.cc_CallContext.currUser.accountId`.

## Outputs

`Map<String, Object>` that includes the `ccrz.ccApiAccount.ACCOUNTS` output from `ccrz.ccApiAccount.fetch`.

SEE ALSO:

[ccrz.ccApiAccount.fetch](#)

[ccrz.ccApiAccount.getAnonymous](#)

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

## ccrz.ccLogicAccountValidateNew

Define the business logic for checking new account records for missing fields.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiAccount.validateNew](#)

## Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method checks that a new account record defines required fields, and returns an error message specific to each missing field.

### ccrz.ccLogicAccountValidateNew.process

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method checks that a new account record defines required fields, and returns an error message specific to each missing field.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiAccount.validateNew` method.

This method requires the following keys in the input data:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiAccount.ACCOUNT_NEW`

`List<String>` that specifies required fields and values for a created [account record](#).

Field	Type
billingAddress	Map<String, Object> (see the following table)
companyName	String
companyType	String
currencyCode	String
firstName	String
lastName	String
language	String
primaryPhone	String
shippingAddress	Map<String, Object> (see the following table)
userName	String

The `billingAddress` and `shippingAddress` maps contain the following required fields:

Field	Required?	Type
address1	Required	String
city	Required	String
postalCode	Required	String

## Logic Performed

This method completes the following actions:

1. Checks for each required field. If a required field is missing, this method returns a `ccrz.cc_bean_Message` error message specific to the missing field.

2. Calls `ccrz.ccApiUser.fetch`, where the `ccrz.ccApiUser.USER_NAME` input key is the value of `userName` from `ccrz.ccApiAccount.ACCOUNT_NEW`, to verify that the user exists.
3. Checks the value of the Redirect To Login configuration of the User Profile (Registration) module (`ur.dirlogin`) in your storefront's configuration settings. When that configuration's value is null, this method verifies the user's password for logging in to the storefront.

## Outputs


Map<String, Object> that can include the following keys:

### `ccrz.ccApi.MESSAGES`

List<`ccrz.cc_bean_Message`>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint.</a>

SEE ALSO:

[Account \(B2B Commerce for Visualforce\)](#)

[ccrz.ccApiUser.fetch](#)

[Salesforce Help: Account, Contact, and User Information for B2B Commerce for Visualforce Buyers](#)

## ccrz.ccLogicAddressCreateAddress

Define the business logic for creating a contact address record from the fields and values that you specify. On the storefront, this logic service provider is invoked by default when a buyer enters new address information during line-level independence (LLI) checkout flow.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiAddress.createAddress](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

#### [processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

#### [processAddress](#)

Constructs a `ccrz__E_ContactAddr__c` sObject from the input values.

#### [createAddress](#)

Executes the DML `insert` operation on the contact address records to create, and sets Salesforce IDs on the records.

#### [refetch](#)

Call the `ccrz.ccApiAddress.fetch` method for fetching the created contact addresses.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### SEE ALSO:

[Salesforce Help Account Address Books and Contact Addresses for B2B Commerce for Visualforce Buyers](#)

[Salesforce Help Standard Checkout versus Line-Level Independence \(LLI\) Checkout](#)

[Salesforce Help Specify Whether a Buyer Can Modify Addresses on the Checkout Page](#)

## ccrz.ccLogicAddressCreateAddress.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiAddress.createAddress` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicAddressCreateAddress` class in the following order:

1. `processInput`
2. `processAddress`
3. `createAddress`
4. `refetch`
5. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicAddressCreateAddress.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```



## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicAddressCreateAddress.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### **ccrz.ccApiAddress.ADDRESSLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the fields and values to use for creating a [contact address record](#).

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicAddressCreateAddress.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicAddressCreateAddress.processAddress

Constructs a `ccrz__E_ContactAddr__c` sObject from the input values.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processAddress (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicAddressCreateAddress.processInput` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiAddress.ADDRESSLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the fields and values to use for creating a [contact address record](#).

**ccrz.ccService.TRANSFORMINDEX**

Map<String, Object> that the `ccrz.ccServiceCart` data service provider generates for defining how the fields in a map convert to sObject data.

**Inputs (Optional)**

If the working map includes the following key, this method evaluates it:

**ccrz.ccApi.SKIP\_OWNERID**

Boolean

Value	Usage
<code>true</code>	Set the Owner field on the new sObject to a default value.
<code>false</code> (default)	Set the Owner field on the new sObject to the value of <code>ccrz.cc_CallContext.currUserId</code> .

**Logic Performed**

This method completes the following actions:

1. For each address represented in `ccrz.ccApiAddress.ADDRESSLIST`, this method evaluates `ccrz.ccService.TRANSFORMINDEX` and serializes the input data to a `ccrz__E_ContactAddr__c` sObject.
2. Evaluates `ccrz.ccApi.SKIP_OWNERID` for setting the Owner field on each `ccrz__E_ContactAddr__c` sObject.

**Outputs**

This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccApiAddress.ADDR\_TO\_INSERT**

List<`ccrz__E_ContactAddr__c`> of the contact address sObject data to insert.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[Salesforce Help Account Address Books and Contact Addresses for B2B Commerce for Visualforce Buyers](#)

**ccrz.ccLogicAddressCreateAddress.createAddress**

Executes the DML `insert` operation on the contact address records to create, and sets Salesforce IDs on the records.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createAddress (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicAddressCreateAddress.processAddress` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiAddress.ADDR_TO_INSERT`


`List<ccrz__E_ContactAddr__c>` of the contact address sObject data to insert.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

### `ccrz.ccApiAddress.ADDRESSIDLIST`

`List<String>` of Salesforce IDs of the created contact address records.

### `ccrz.ccApiAddress.ADDR_TO_INSERT`

`List<ccrz__E_ContactAddr__c>` of the inserted contact address sObject records with Salesforce IDs set.

SEE ALSO:

[Inserting and Updating Records](#)

## `ccrz.ccLogicAddressCreateAddress.refetch`

Call the `ccrz.ccApiAddress.fetch` method for fetching the created contact addresses.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)


This method accepts the `Map<String, Object>` after the `ccrz.ccLogicAddressCreateAddress.createAddress` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApi.SIZING**

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

### **ccrz.ccApiAddress.ADDR\_TO\_INSERT**

`List<ccrz__E_ContactAddr__c>` of the inserted contact address sObject records with Salesforce IDs set.

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiAddress.fetch` outputs.

SEE ALSO:

[ccrz.ccApiAddress.fetch](#)

## ccrz.ccLogicAddressCreateAddress.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicAddressCreateAddress.refetch` method completes.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApi.SIZING`
- `ccrz.ccApi.SKIP_OWNERID`
- `ccrz.ccApiAddress.ADDR_TO_INSERT`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccService.TRANSFORMINDEX`

## ccrz.ccLogicAddressBookCreateAddressBook

Define the business logic for creating an account address book record from the fields and values that you specify.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiAddressBook.createAddressBook](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

#### [processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

#### [processAddressBook](#)

Constructs a `ccrz__E_AccountAddressBook__c` sObject from the input values.

#### [createAddressBook](#)

Executes the DML `insert` operation on the account address book records to create, and sets Salesforce IDs on the records.

#### [refetch](#)

Calls the `ccrz.ccApiAddressBook.fetch` method for fetching the created account address books.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### SEE ALSO:

[Salesforce Help Account Address Books and Contact Addresses for B2B Commerce for Visualforce Buyers](#)

## ccrz.ccLogicAddressBookCreateAddressBook.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiAddressBook.createAddressBook` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicAddressBookCreateAddressBook` class in the following order:

1. `processInput`
2. `processAddressBook`
3. `createAddressBook`
4. `refetch`
5. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicAddressBookCreateAddressBook.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicAddressBookCreateAddressBook.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### **ccrz.ccApiAddressBook.ADDRESSBOOKS**

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the fields and values to use for creating an [account address book record](#).

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicAddressBookCreateAddressBook.prepReturn method` removes this intermediate key from the working map.

## ccrz.ccLogicAddressBookCreateAddressBook.processAddressBook

Constructs a `ccrz__E_AccountAddressBook__c` sObject from the input values.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



## Signature

```
global virtual Map<String, Object> processAddressBook (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicAddressBookCreateAddressBook.processInput` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiAddressBook.ADDRESSBOOKS**

`List<Map<String, Object>>`, where each `Map<String, Object>` contains the fields and values to use for creating an [account address book record](#).

### **ccrz.ccService.TRANSFORMINDEX**

`Map<String, Object>` that the `ccrz.ccServiceCart` data service provider generates for defining how the fields in a map convert to sObject data.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### **ccrz.ccApi.SKIP\_OWNERID**

Boolean

Value	Usage
<i>true</i>	Set the Owner field on the new sObject to a default value.
<i>false</i> (default)	Set the Owner field on the new sObject to the value of <code>ccrz.cc_CallContext.currUserId</code> .

## Logic Performed

This method completes the following actions:

1. For each address book represented in `ccrz.ccApiAddressBook.ADDRESSBOOKS`, this method evaluates `ccrz.ccService.TRANSFORMINDEX` and serializes the input data to a `ccrz__E_AccountAddressBook__c` sObject.
2. Evaluates `ccrz.ccApi.SKIP_OWNERID` for setting the Owner field on each `ccrz__E_AccountAddressBook__c` sObject.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiAddressBook.AB\_TO\_INSERT**

List<[ccrz\\_\\_E\\_AccountAddressBook\\_\\_c](#)> of the account address book sObject data to insert.

SEE ALSO:

[ccrz\\_\\_E\\_AccountAddressBook\\_\\_c](#)

[Salesforce Help Account Address Books and Contact Addresses for B2B Commerce for Visualforce Buyers](#)

**ccrz.ccLogicAddressBookCreateAddressBook.createAddressBook**

Executes the DML `insert` operation on the account address book records to create, and sets Salesforce IDs on the records.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> createAddressBook (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working `Map<String, Object>` after the [ccrz.ccLogicAddressBookCreateAddressBook.processAddressBook](#) method completes.

This method requires the following key in the working map:

**ccrz.ccApiAddressBook.AB\_TO\_INSERT**

List<[ccrz\\_\\_E\\_AccountAddressBook\\_\\_c](#)> of the account address book sObject data to insert.

**Outputs**

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiAddressBook.AB\_TO\_INSERT**

List<[ccrz\\_\\_E\\_AccountAddressBook\\_\\_c](#)> of the inserted account address book sObject records with Salesforce IDs set.

SEE ALSO:

[ccrz\\_\\_E\\_AccountAddressBook\\_\\_c](#)

[Inserting and Updating Records](#)

**ccrz.ccLogicAddressBookCreateAddressBook.refetch**

Calls the `ccrz.ccApiAddressBook.fetch` method for fetching the created account address books.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

**Inputs (Required)**


This method accepts the working `Map<String, Object>` after the [ccrz.ccLogicAddressBookCreateAddressBook.createAddressBook](#) method completes.

This method requires the following keys in the working map:

**ccrz.ccApi.SIZING**

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAddressBook.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

**ccrz.ccApiAddressBook.AB\_TO\_INSERT**

List<[ccrz\\_\\_E\\_AccountAddressBook\\_\\_c](#)> of the inserted account address book sObject records with Salesforce IDs set.

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiAddressBook.fetch` outputs.

SEE ALSO:

[ccrz.ccApiAddressBook.fetch](#)

## ccrz.ccLogicAddressBookCreateAddressBook.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicAddressBookCreateAddressBook.refetch` method completes.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApi.SIZING`
- `ccrz.ccApi.SKIP_OWNERID`
- `ccrz.ccApiAddressBook.AB_TO_INSERT`
- `ccrz.ccLogic.ORIGINAL_INPUT`

- `ccrz.ccService.TRANSFORMINDEX`

## ccrz.ccLogicAddressBookRemoveAddressBook

Define the business logic for deleting account address book records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiAddressBook.removeAddressBook](#)

### Methods

#### IN THIS SECTION:

##### [process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

##### [processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

##### [fetchAddressBook](#)

Calls the `ccrz.ccApiAddressBook.fetch` method to return the account address book records that you want to delete.

##### [removeAddressBook](#)

Executes the DML `delete` operation for the account address book records.

##### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

#### SEE ALSO:

[Salesforce Help Account Address Books and Contact Addresses for B2B Commerce for Visualforce Buyers](#)

## ccrz.ccLogicAddressBookRemoveAddressBook.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiAddressBook.removeAddressBook` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicAddressBookRemoveAddressBook` class in the following order:

1. `processInput`
2. `fetchAddressBook`
3. `removeAddressBook`
4. `prepReturn`

### Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicAddressBookRemoveAddressBook.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicAddressBookRemoveAddressBook.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### **ccrz.ccApiAddressBook.IDS**


`List<String>` of Salesforce IDs of the [account address book records](#) to delete.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint.</a>

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicAddressBookRemoveAddressBook.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicAddressBookRemoveAddressBook.fetchAddressBook

Calls the `ccrz.ccApiAddressBook.fetch` method to return the account address book records that you want to delete.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchAddressBook (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicAddressBookRemoveAddressBook.processInput` method completes.

This method requires the following key in the working map:

#### **ccrz.ccApiAddressBook.IDS**

`List<String>` of Salesforce IDs of the [account address book records](#) to delete.

### Logic Performed

This method completes the following actions:

1. Passes `ccrz.ccApiAddressBook.IDS` to `ccrz.ccApiAddressBook.fetch`. By default, this query applies the following [data size request parameters](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAddressBook.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XS,
        ccrz.ccApi.SZ_SKIPTRZ => true
    }
}
```

2. Parses the `ccrz.ccApiAddressBook.fetch` return data for the returned account address books.



## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiAddressBook.AB\_TO\_REMOVE**

`List<ccrz__E_AccountAddressBook__c>` of the account address book records to delete.

SEE ALSO:

[ccrz.ccApiAddressBook.fetch](#)

## ccrz.ccLogicAddressBookRemoveAddressBook.removeAddressBook

Executes the DML `delete` operation for the account address book records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> removeAddressBook (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the

[ccrz.ccLogicAddressBookRemoveAddressBook.fetchAddressBook](#) method completes.

This method requires the following key in the working map:

**ccrz.ccApiAddressBook.AB\_TO\_REMOVE**


`List<ccrz__E_AccountAddressBook__c>` of the account address book records to delete.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

SEE ALSO:

[Deleting Records](#)

## ccrz.ccLogicAddressBookRemoveAddressBook.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicAddressBookRemoveAddressBook.removeAddressBook` method completes.

### Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.


- `ccrz.ccApi.SIZING`
- `ccrz.ccApiAddressBook.AB_TO_REMOVE`
- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicAttrFetch

Define the business logic for filtering `ccrz__E_Attribute__c` records returned from the `ccrz.ccApiAttribute.fetch` method's SOQL query. This class also defines whether `ccrz__E_Attribute__c` return data is keyed by `ccrz__AttributeID__c` or Salesforce ID.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicAttrFetch](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiAttribute.fetch](#)

## ccrz.ccLogicCartAddTo

Define the business logic for adding products to a cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiCart.addTo](#)

### Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

[processInputData](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [fetchCart](#)

Calls the `ccrz.ccApiCart.fetch` method with `ccrz.ccApi.SZ_XL` data sizing for fetching the active cart to add products to.

#### [addCouponCode](#)

Calls the `ccrz.ccApiCoupon.apply` method for applying a coupon code to the cart.

#### [addMajorLineItems](#)

Reads the input line data for specifying the cart's major line items, such as standard products or composite products.

#### [addMinorLineItems](#)

Reads the input line data for specifying the cart's minor line items, such as component products.

#### [addCurrentLineDataToCart](#)

Calls other inner, non-overrideable methods that sanitize and update the line data for adding cart items.

#### [addCartLinesToItems](#)

Loops through `ccrz.ccApiCart.WORKING_LINE_DATAS` and creates cart items.

#### [preIterationData](#)

Begins the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by evaluating the product information for the current line item.

#### [setCartItemFields](#)

Continues the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by setting values on the current line item's `ccrz__E_CartItem__c` record.

#### [handlePricing](#)

Continues the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by calling the appropriate inner method for pricing the current line item based on its product type.

#### [priceBasicProductItem](#)

Sets a price for a major cart item that's a standard product or an assembly product.

#### [priceBundleItem](#)

Sets a price for a major cart item that's a bundle product.

#### [priceKitItem](#)

Sets the price for a major cart item that's a kit or dynamic kit to 0.00. The buyer's entitled price for a kit or dynamic kit depends on the prices of its individual component products.

#### [applyTieredPricing](#)

Applies a tier-based pricing adjustment, if applicable, to a major cart item.

#### [priceAttributeProductItem](#)

Sets a price for a major cart item that's an attribute-based product.

#### [priceExternalItem](#)

Sets an externally defined price for a major cart item.

#### [validateItem](#)

Continues the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by returning exceptions for an invalid major cart item.

#### [setSubAmount](#)

Continues the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by setting the subtotal price for the total quantity of a major cart item.

[concludeItemAdd](#)

Completes the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by finalizing fields and relationships on the cart item.

[upsertCartItems](#)

Executes the DML `upsert` operation for the created cart item records, which assigns IDs to each record.

[concludeAddTo](#)

Returns final data keys, resets the cart's validation status, and optionally reprices and refetches cart records.

## ccrz.ccLogicCartAddTo.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.addTo` method.

### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartAddTo` class in the following order:

1. `processInputData`
2. `fetchCart`
3. `addCouponCode`
4. `addMajorLineItems`, which calls into `addCurrentLineDataToCart`
5. `addMinorLineItems`, which calls into `addCurrentLineDataToCart`
6. `addCurrentLineDataToCart`, which calls into `addCartLinesToItems`
7. `addCartLinesToItems`, which executes a For loop that calls into the following methods for each item:

- a. `prepIterationData`
  - b. `setCartItemFields`
  - c. `handlePricing`, which calls the appropriate inner method for pricing the current line item based on its product type:
    - `priceBasicProductItem`
    - `priceBundleItem`
    - `priceKitItem`
    - `priceDynamicKitItem`, which calls back into `priceKitItem`
    - `applyTieredPricing`
    - `priceAttributeProductItem`
    - `priceExternalItem`
  - d. `validateItem`
  - e. `setSubAmount`
  - f. `concludeItemAdd`
8. `upsertCartItems`
  9. `concludeAddTo`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. The method adds the following intermediate key to this map:

### `ccrz.ccApiCart.RESPONSE`

`Map<String, Object>` that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## `ccrz.ccLogicCartAddTo.processInputData`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInputData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.process` method.

This method checks whether the input map specifies any of the following keys, which determine this method's output keys and affect other downstream logic methods.

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of an existing cart record to add products to.

### `ccrz.ccApiCart.CART_ENCID`


String that specifies the encrypted ID of an existing cart record to add products to.

### `ccrz.ccApiCart.COUPON_CODE`

String that specifies a coupon code to apply to the cart.

### `ccrz.ccApiCart.ISREPRICE`

Boolean

Value	Usage
<code>true</code>	Reprice the cart after the items are added.  <b>Note:</b> This operation can be expensive, depending on the complexity of your cart data, but ensures that prices are current and entitlement is correct.
<code>false</code> (default)	Don't reprice cart items.

### `ccrz.ccApiCart.ISSKIPPRICING`

Boolean

Value	Usage
<code>true</code>	Call the <code>ccrz.ccLogicProductPricing</code> class for pricing the products you're adding to the cart.
<code>false</code> (default)	Ignore pricing information for the products you're adding to the cart.

### `ccrz.ccApiCart.LINE_DATA`

`List<Map<String, Object>>`, where each `Map<String, Object>` describes a `ccrz__E_CartItem__crecord` to create. For information about this list's keys and requirements, see [Inputs for Cart Line Data](#).

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiCart.RESPONSE**

Map<String, Object> that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key. This method instantiates the following keys with initially empty values, which subsequent methods populate:

**ccrz.ccApiCart.INCRERROR**

Map<String, Decimal>

**ccrz.ccApiCart.INCRWARN**

Map<String, Decimal>

**ccrz.ccApiCart.INELIGIBLEPRODUCTS**

List<String>

**ccrz.ccApiCart.NONSTDPRODUCTS**

List<String>

**ccrz.ccApiCart.PROSDADDEDTOCART**

Map<ccrz\_\_E\_CartItem\_\_c, String>

**ccrz.ccApiCart.HAS\_COUPON\_CODE**

Boolean

Value	Usage
true	The input map specified <code>ccrz.ccApiCart.COUPON_CODE</code> .
false	The input map didn't specify <code>ccrz.ccApiCart.COUPON_CODE</code> .

**ccrz.ccApiCart.HAS\_ID**

Boolean

Value	Usage
true	The input map specified either <code>ccrz.ccApiCart.CART_ID</code> or <code>ccrz.ccApiCart.CART_ENCID</code> .
false	The input map didn't specify a cart ID.

**ccrz.ccApiCart.HAS\_LINE\_DATA**

Boolean

Value	Usage
true	The input map specified <code>ccrz.ccApiCart.LINE_DATA</code> .
false	The input map didn't specify <code>ccrz.ccApiCart.LINE_DATA</code> .

**ccrz.ccApiCart.IS\_ADD\_SEPARATE**

Boolean



Value	Usage
true	The Add Line Items Separately configuration of the Quick Order module is TRUE.
false	The Add Line Items Separately configuration of the Quick Order module is FALSE.

**!** **Important:** If the input map specifies `ccrz.ccApiCart.COUPON_CODE`, make sure that this key evaluates to `false`. The Add Line Items Separately configuration of the Quick Order module isn't compatible with coupons.

#### **ccrz.ccApiCart.IS\_SELLERS\_ENABLED**

Boolean

Value	Usage
true	The Enabled configuration of the Sellers module is TRUE.
false	The Enabled configuration of the Sellers module is FALSE.

#### **ccrz.ccApiCart.IS\_SKIP\_PRICING**

Boolean

Value	Usage
true	The input map set <code>ccrz.ccApiCart.ISSKIPPRICING</code> => <code>true</code> to skip pricing logic.
false	The input map set <code>ccrz.ccApiCart.ISSKIPPRICING</code> => <code>false</code> , or didn't specify a value for <code>ccrz.ccApiCart.ISSKIPPRICING</code> .

#### **ccrz.ccApiCart.ISREPRICE**

Boolean

Value	Usage
true	The input map set <code>ccrz.ccApiCart.ISREPRICE</code> => <code>true</code> .
false	The input map set <code>ccrz.ccApiCart.ISREPRICE</code> => <code>false</code> , or didn't specify a value for <code>ccrz.ccApiCart.ISREPRICE</code> .

SEE ALSO:

[Salesforce Help Offer Coupons for Discounts on Storefront Purchases](#)

## **ccrz.ccLogicCartAddTo.fetchCart**

Calls the `ccrz.ccApiCart.fetch` method with `ccrz.ccApi.SZ_XL` data sizing for fetching the active cart to add products to.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchCart (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.processInputData` method.

This method evaluates the following values in the working map:

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of an existing cart record to add products to.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of an existing cart record to add products to.

### `ccrz.ccApiCart.HAS_ID`

Boolean

Value	Usage
<i>true</i>	The input map specified either <code>ccrz.ccApiCart.CART_ID</code> or <code>ccrz.ccApiCart.CART_ENCID</code> .
<i>false</i>	The input map didn't specify a cart ID.

### `ccrz.ccApiCart.HAS_LINE_DATA`

Boolean

Value	Usage
<i>true</i>	The input map specified <code>ccrz.ccApiCart.LINE_DATA</code> .
<i>false</i>	The input map didn't specify <code>ccrz.ccApiCart.LINE_DATA</code> .

**ccrz.ccApiCart.RESPONSE**

Map<String, Object> that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

The values of these keys determine how this method executes:

1. If `ccrz.ccApiCart.HAS_ID` is *true* and `ccrz.ccApiCart.HAS_LINE_DATA` is *true*, the method tries to fetch the specified cart.
2. If `ccrz.ccApiCart.HAS_ID` is *false* but `ccrz.ccApiCart.HAS_LINE_DATA` is *true*, the method tries to fetch the active cart for the current user and storefront, based on `ccrz.cc_CallContext`.



**Note:** In B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12), this evaluation results in an exception when the active cart's `ccrz__CartStatus__c` value is `Open`. This exception occurs because the cart can currently be subject to pricing, placing an order, or requiring repair.

3. Otherwise, if the method can't determine an active cart to add products to, the method calls the `ccrz.ccApiCart.create` method.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApiCart.CARTITEMSBYID**

Map<String, List<ccrz\_\_E\_CartItem\_\_c>> for the fetched `ccrz__E_Cart__c`.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[ccrz.ccApiCart.create](#)

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCartAddTo.addCouponCode

Calls the `ccrz.ccApiCoupon.apply` method for applying a coupon code to the cart.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> addCouponCode (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.fetchCart` method.

This method executes only if the working map includes the following key with a value of `true`:

### **ccrz.ccApiCart.HAS\_COUPON\_CODE**

Boolean

Value	Usage
<code>true</code>	The input map specified <code>ccrz.ccApiCart.COUPON_CODE</code> .
<code>false</code>	The input map didn't specify <code>ccrz.ccApiCart.COUPON_CODE</code> .

This method also evaluates the following keys:

### **ccrz.ccApiCart.RESPONSE**

`Map<String, Object>` that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

This method passes the `ccrz.ccApiCart.COUPON_CODE` and `ccrz.ccApiCart.CART_ENCID` keys to the [ccrz.ccApiCoupon.apply method](#).

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiCart.ISREPRICE**

Boolean that's set to `true` after the coupon is applied for repricing the cart.

SEE ALSO:

[ccrz.ccApiCoupon.apply](#)

## ccrz.ccLogicCartAddTo.addMajorLineItems

Reads the input line data for specifying the cart's major line items, such as standard products or composite products.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> addMajorLineItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.addCouponCode` method.

This method requires the following key in the working map:

### **ccrz.ccApiCart.LINE\_DATA**

`List<Map<String, Object>>`, where each `Map<String, Object>` describes a `ccrz__E_CartItem__c` record to create for a major line item. For information about this list's keys and requirements, see [Inputs for Cart Line Data](#).

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiCart.WORKING\_LINE\_DATAS**

`List<Map<String, Object>>` of working map line data, initialized with a copy of the input `ccrz.ccApiCart.LINE_DATA`.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartAddTo.addMinorLineItems

Reads the input line data for specifying the cart's minor line items, such as component products.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> addMinorLineItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.addMajorLineItems` method.

This method requires the following key in the working map:

### **ccrz.ccApiCart.MINOR\_LINE\_DATAS**

`List<Map<String, Object>>`, where each `<Map<String, Object>>` contains `ccrz.ccApiCart.LINE_DATA` fields for a component product of a composite product. This key is a copy of the `ccrz.ccApiCart.LINE_DATA_MINOR_LINE_DATAS` key from the original input map.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiCart.WORKING\_LINE\_DATAS**

`List<Map<String, Object>>` of working map line data, updated with a copy of the input `ccrz.ccApiCart.MINOR_LINE_DATAS`.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartAddTo.addCurrentLineDataToCart

Calls other inner, non-overridable methods that sanitize and update the line data for adding cart items.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> addCurrentLineDataToCart (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.addMajorLineItems` or `ccrz.ccLogicCartAddTo.addMinorLineItems` method.

This method passes the working map through a chain of inner, non-overridable methods that:

- Remove duplicate any line data
- Fetch the product details that correspond to the major and minor line items
- Fetch details for externally priced products
- Evaluate the `ccrz.ccApiCart.IS_ADD_SEPARATE` key for determining whether to combine multiple lines for the same SKU

## Outputs

This method returns the working `Map<String, Object>` after the inner methods complete.

## ccrz.ccLogicCartAddTo.addCartLinesToItems

Loops through `ccrz.ccApiCart.WORKING_LINE_DATAS` and creates cart items.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> addCartLinesToItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.addCurrentLineDataToCart` method.

This method requires the following key in the working map:

### **ccrz.ccApiCart.WORKING\_LINE\_DATAS**

`List<Map<String, Object>>` of major and minor line items to create `ccrz__E_CartItem__c` records for.

This method executes a For loop that over each line item, and invokes inner methods that set values for the corresponding cart item.

#### 1. `prepIterationData`

2. [setCartItemFields](#)
3. [handlePricing](#)
4. [validateItem](#)
5. [setSubAmount](#)
6. [concludeItemAdd](#)

After the For loop completes, this method calls the [upsertCartItems](#) method for upserting the final cart item records.

## Outputs

This method returns the working `Map<String, Object>` after the inner loop methods complete.

For each loop, this method adds the following intermediate key:

### **ccrz.ccApiCart.CURR\_LINE\_DATA**

`Map<String, Object>` that describes the line data for only the line currently being evaluated within the loop.

For each loop, this method also removes the following intermediate keys:

### **ccrz.ccApiCart.CURR\_CARTITEM**

`ccrz__E_CartItem__c` that corresponded to the item from the previous loop.

### **ccrz.ccApiCart.CURR\_PRODUCT**

`Map<String, Object>` that describes the `ccrz__E_Product__c` that corresponded to the item from the previous loop.

### **ccrz.ccApiCart.CURR\_PRODUCT\_KEY**

String that specifies the product SKU or ID that corresponded to the item from the previous loop.

### **ccrz.ccApiCart.CURR\_PRODUCT\_QUANTITY\_RULE\_RESULT**

`Map<String, Object>` that describes the result of the evaluated property quantity rules for the item from the previous loop.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartAddTo.preIterationData

Begins the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by evaluating the product information for the current line item.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



## Signature

```
global virtual Map<String, Object> prepIterationData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method.

This method passes the working map through a chain of inner, non-overridable methods that:

- Determines the product that corresponds to the current item
- Determines the current item's product type, and returns exceptions for unsupported product configurations
- Checks for any [product quantity rules](#) and returns warnings or errors, if applicable

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiCart.CURR_CARTITEM`

`ccrz__E_CartItem__c` that corresponds to the current item.

### `ccrz.ccApiCart.CURR_PRODUCT`

`Map<String, Object>` that describes the `ccrz__E_Product__c` that corresponds to the current item.

### `ccrz.ccApiCart.CURR_PRODUCT_QUANTITY_RULE_RESULT`

`Map<String, Object>` that describes the result of the evaluated property quantity rules.

### `ccrz.ccApiCart.IS_PART_OF_BUNDLE`

Boolean

Value	Usage
true	The current item is a component product in a <a href="#">bundle</a> .
false	The current item isn't part of a bundle.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## `ccrz.ccLogicCartAddTo.setCartItemFields`

Continues the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by setting values on the current line item's `ccrz__E_CartItem__c` record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> setCartItemFields(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.prepIterationData` method.

This method requires the following keys in the working map:

### `ccrz.ccApiCart.CURR_CARTITEM`

`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

### `ccrz.ccApiCart.CURR_LINE_DATA`

`Map<String, Object>` that describes the line data for only the line currently being evaluated within the loop.

### `ccrz.ccApiCart.CURR_PRODUCT`

`Map<String, Object>` that describes the `ccrz__E_Product__c` that corresponds to the item currently being evaluated within the loop.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following fields on the `ccrz__E_CartItem__c` record:

<code>ccrz__E_CartItem__c</code> Field	Source for Value
<code>ccrz__ItemLabel__c</code>	<code>ccrz.ccApiCart.CURR_LINE_DATA</code> input key's <code>ccrz.ccApiCart.LINE_DATA_LABEL</code> subkey
<code>ccrz__PrimaryAttr__c</code>	<code>ccrz.ccApiCart.CURR_LINE_DATA</code> input key's <code>ccrz.ccApiCart.LINE_DATA_ATTRIBUTE_LIST</code> subkey
<code>ccrz__Product__c</code>	<code>ccrz.ccApiCart.CURR_PRODUCT</code> input key's <code>sfid</code> field
<code>ccrz__ProductType__c</code>	<code>ccrz.ccApiCart.CURR_PRODUCT</code> input key's <code>productType</code> field
<code>ccrz__RequestDate__c</code>	<code>Date.today</code>

ccrz__E_CartItem__c Field	Source for Value
ccrz__SecondaryAttr__c	ccrz.ccApiCart.CURR_LINE_DATA input key's ccrz.ccApiCart.LINE_DATA_ATTRIBUTE_LIST subkey
ccrz__SelectedAttr__c	ccrz.ccApiCart.CURR_LINE_DATA input key's ccrz.ccApiCart.LINE_DATA_ATTRIBUTE_LIST subkey
ccrz__SortName__c  <b>Note:</b> This field is set only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).	ccrz.ccApiCart.CURR_PRODUCT input key's name, in the following order of precedence: <ol style="list-style-type: none"><li>1. Localized name field</li><li>2. Immediate name field</li><li>3. alternateName field</li></ol>
ccrz__StoreId__c	<a href="#">ccrz.cc_CallContext.storefront</a>
ccrz__TertiaryAttr__c	ccrz.ccApiCart.CURR_LINE_DATA input key's ccrz.ccApiCart.LINE_DATA_ATTRIBUTE_LIST subkey
ccrz__UnitOfMeasure__c	ccrz.ccApiCart.CURR_PRODUCT input key's UOM field

 **Note:** For information about the line data parameters, see [Inputs for Cart Line Data](#).

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartAddTo.handlePricing

Continues the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by calling the appropriate inner method for pricing the current line item based on its product type.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> handlePricing(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.setCartItemFields` method.

This method executes only if the working map includes the following keys, each with a value of `false`:

### `ccrz.ccApiCart.IS_PART_OF_BUNDLE`

Boolean

Value	Usage
<code>true</code>	The current item is a component product in a <a href="#">bundle</a> . The bundle's composite product—a major line item in the cart—specifies the bundle's price, so pricing is skipped for this minor line item.
<code>false</code>	The current item isn't part of a bundle.

### `ccrz.ccApiCart.IS_SKIP_PRICING`

Boolean

Value	Usage
<code>true</code>	The input map set <code>ccrz.ccApiCart.ISSKIPPRICING =&gt; true</code> to skip pricing logic.
<code>false</code>	The input map set <code>ccrz.ccApiCart.ISSKIPPRICING =&gt; false</code> , or didn't specify a value for <code>ccrz.ccApiCart.ISSKIPPRICING</code> .

This method also evaluates the following keys:

### `ccrz.ccApiCart.CURR_CARTITEM`

`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

### `ccrz.ccApiCart.CURR_LINE_DATA`

`Map<String, Object>` that describes the line data for only the line currently being evaluated within the loop.

## Outputs

This method returns the working `Map<String, Object>` after the appropriate inner pricing method completes.

Product Type	Method
Standard product or assembly	<code>priceBasicProductItem</code>
Bundle	<code>priceBundleItem</code>
Kit	<code>priceKitItem</code>
Dynamic Kit	<code>priceDynamicKitItem</code> , which calls into <code>priceKitItem</code>

Product Type	Method
Aggregated	None, because an aggregated product's composite product price list item is always 0.00.
Tiered pricing	<a href="#">applyTieredPricing</a>
Attribute-based product	<a href="#">priceAttributeProductItem</a>
Externally priced product	<a href="#">priceExternalItem</a>

**SEE ALSO:**

[Salesforce Help Types of Products for B2B Commerce for Visualforce](#)

## ccrz.ccLogicCartAddTo.priceBasicProductItem

Sets a price for a major cart item that's a standard product or an assembly product.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> priceBasicProductItem (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.handlePricing` method.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CURR\_CARTITEM**

`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

**ccrz.ccApiCart.CURR\_LINE\_DATA**

`Map<String, Object>` that describes the line data for only the line currently being evaluated within the loop.

**ccrz.ccApiCart.CURR\_PRODUCT**

Map<String, Object> that describes the `ccrz__E_Product__c` that corresponds to the item currently being evaluated within the loop.

**ccrz.ccApiCart.IS\_SELLERS\_ENABLED**

Boolean

Value	Usage
<code>true</code>	The Enabled configuration of the Sellers module is TRUE.
<code>false</code>	The Enabled configuration of the Sellers module is FALSE.

**ccrz.ccApiCart.PRODUCTS\_PRICING\_DATA**

Map<String, Object> of product pricing data, keyed by product ID. The non-overridable `ccrz.ccLogicCartAddTo.fetchProducts` method added this key to the working map.

## Outputs

This method returns the working Map<String, Object>, and sets the following fields on the `ccrz__E_CartItem__c` record:

<code>ccrz__E_CartItem__c</code> Field	Source for Value
<code>ccrz__Price__c</code>	The buyer's entitled price for the product.
<code>ccrz__RecurringPrice__c</code>	The recurring subscription price for the product, if applicable.

## SEE ALSO:

[Salesforce Help Create a Product for Your Storefront](#)

[Salesforce Help Create an Assembly Product](#)

## ccrz.ccLogicCartAddTo.priceBundleItem

Sets a price for a major cart item that's a bundle product.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> priceBundleItem(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.handlePricing` method.

This method requires the following keys in the working map:

### `ccrz.ccApiCart.CURR_CARTITEM`

`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

### `ccrz.ccApiCart.CURR_PRODUCT`

`Map<String, Object>` that describes the `ccrz__E_Product__c` that corresponds to the item currently being evaluated within the loop.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following fields on the `ccrz__E_CartItem__c` record:

<code>ccrz__E_CartItem__c</code> Field	Source for Value
<code>ccrz__Price__c</code>	The buyer's entitled price for the bundle.
<code>ccrz__RecurringPrice__c</code>	The recurring subscription price for the bundle, if applicable.

SEE ALSO:

[Salesforce Help Create a Bundle](#)

## `ccrz.ccLogicCartAddTo.priceKitItem`

Sets the price for a major cart item that's a kit or dynamic kit to 0.00. The buyer's entitled price for a kit or dynamic kit depends on the prices of its individual component products.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> priceKitItem(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.handlePricing` method.

This method requires the following keys in the working map:

### `ccrz.ccApiCart.CURR_CARTITEM`

`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

### `ccrz.ccApiCart.CURR_LINE_DATA`

`Map<String, Object>` that describes the line data for only the line currently being evaluated within the loop.

### `ccrz.ccApiCart.CURR_PRODUCT`

`Map<String, Object>` that describes the `ccrz__E_Product__c` that corresponds to the item currently being evaluated within the loop.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following fields on the `ccrz__E_CartItem__c` record:

<code>ccrz__E_CartItem__c</code> Field	Source for Value
<code>ccrz__Price__c</code>	<code>0.00</code>
<code>ccrz__RecurringPrice__c</code>	<code>0.00</code>

SEE ALSO:

[Salesforce Help Create a Kit](#)

## `ccrz.ccLogicCartAddTo.applyTieredPricing`

Applies a tier-based pricing adjustment, if applicable, to a major cart item.

## Compatibility

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> applyTieredPricing(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.handlePricing` method.

This method requires the following keys in the working map:

### `ccrz.ccApiCart.CURR_CARTITEM`

`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

### `ccrz.ccApiCart.CURR_LINE_DATA`

`Map<String, Object>` that describes the line data for only the line currently being evaluated within the loop.

### `ccrz.ccApiCart.CURR_PRODUCT`

`Map<String, Object>` that describes the `ccrz__E_Product__c` that corresponds to the item currently being evaluated within the loop.

### `ccrz.ccApiCart.PRODUCTS_PRICING_DATA`

`Map<String, Object>` of product pricing data, keyed by product ID. The non-overrideable `ccrz.ccLogicCartAddTo.fetchProducts` method added this key to the working map.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following fields on the `ccrz__E_CartItem__c` record:

<code>ccrz__E_CartItem__c</code> Field	Source for Value
<code>ccrz__Price__c</code>	The price evaluated from the product's pricing tiers, based on the quantity of the item in the cart.

SEE ALSO:

[Salesforce Help Adjust Product Prices Based on Tiered Quantities](#)

## ccrz.ccLogicCartAddTo.priceAttributeProductItem

Sets a price for a major cart item that's an attribute-based product.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> priceAttributeProductItem (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.handlePricing` method.

This method requires the following keys in the working map:

#### **ccrz.ccApiAttribute.ATTRIBUTES**

`Map<String, ccrz__E_Attribute__c>` returned from a `ccrz.ccApiAttribute.fetch` query.

#### **ccrz.ccApiCart.CURR\_CARTITEM**

`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

#### **ccrz.ccApiCart.CURR\_LINE\_DATA**

`Map<String, Object>` that describes the line data for only the line currently being evaluated within the loop.

#### **ccrz.ccApiCart.CURR\_PRODUCT**

`Map<String, Object>` that describes the `ccrz__E_Product__c` that corresponds to the item currently being evaluated within the loop.

#### **ccrz.ccApiCart.PRODUCTS\_PRICING\_DATA**

`Map<String, Object>` of product pricing data, keyed by product ID. The non-overridable `ccrz.ccLogicCartAddTo.fetchProducts` method added this key to the working map.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following fields on the `ccrz__E_CartItem__c` record:

ccrz__E_CartItem__c Field	Source for Value
ccrz__Price__c	The price based on parsing the attribute pricing JSON data.

## ccrz.ccLogicCartAddTo.priceExternalItem

Sets an externally defined price for a major cart item.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> priceExternalItem(Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.handlePricing` method.

This method requires the following keys in the working map:

#### **ccrz.ccApiCart.CURR\_CARTITEM**

`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

#### **ccrz.ccApiCart.CURR\_LINE\_DATA**

`Map<String, Object>` that describes the line data for only the line currently being evaluated within the loop.

From `ccrz.ccApiCart.CURR_LINE_DATA`, this method evaluates the following subkeys:

#### **ccrz.ccApiCart.LINE\_DATA\_EXT\_NAME**

String that specifies a value for the `ccrz__ExtName__c` field on the created `ccrz__E_CartItem__c` record.

#### **ccrz.ccApiCart.LINE\_DATA\_EXT\_SKU**

String that specifies a value for the `ccrz__ExtSKU__c` field on the created `ccrz__E_CartItem__c` record.

#### **ccrz.ccApiCart.LINE\_DATA\_SUBTOTAL**

Decimal that specifies a unique subtotal price for the quantity of externally priced products.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following fields on the `ccrz__E_CartItem__c` record:

<code>ccrz__E_CartItem__c</code> Field	Source for Value
<code>ccrz__ExtName__c</code>	<code>ccrz.ccApiCart.CURR_LINE_DATA</code> input key's <code>ccrz.ccApiCart.LINE_DATA_EXT_NAME</code> subkey
<code>ccrz__ExtSKU__c</code>	<code>ccrz.ccApiCart.CURR_LINE_DATA</code> input key's <code>ccrz.ccApiCart.LINE_DATA_EXT_SKU</code> subkey
<code>ccrz__Price__c</code>	<code>ccrz.ccApiCart.CURR_LINE_DATA</code> input key's <code>ccrz.ccApiCart.LINE_DATA_PRICE</code> subkey
<code>ccrz__PricingType__c</code>	<i>external</i>
<code>ccrz__Subtotal__c</code>	<code>ccrz.ccApiCart.CURR_LINE_DATA</code> input key's <code>ccrz.ccApiCart.LINE_DATA_SUBTOTAL</code> subkey

SEE ALSO:

[Salesforce Help Set Product Prices in the Cart from an External Source](#)

## `ccrz.ccLogicCartAddTo.validateItem`

Continues the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by returning exceptions for an invalid major cart item.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> validateItem(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.handlePricing` method.

This method requires the following key in the working map:

**ccrz.ccApiCart.CURR\_CARTITEM**  
`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

The default implementation of this method returns the following exceptions:

**ccrz.ccApiCart.NonStandardProductException**  
 The major cart item's Product Type is Aggregated.

**ccrz.ccApiCart.InvalidProductException**  
 The major cart item's `ccrz__Price__c` field is null.

## ccrz.ccLogicCartAddTo.setSubAmount

Continues the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by setting the subtotal price for the total quantity of a major cart item.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> setSubAmount (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.validateItem` method.


This method requires the following key in the working map:

**ccrz.ccApiCart.CURR\_CARTITEM**  
`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following fields on the `ccrz__E_CartItem__c` record:

<code>ccrz__E_CartItem__c</code> Field	Source for Value
<code>ccrz__RecurringPriceSubAmt__c</code>	The cart item's recurring subscription price, if applicable, multiplied by the cart item's quantity.
<code>ccrz__SubAmount__c</code>	The cart item's entitled price multiplied by the cart item's quantity.
<code>ccrz__SortPrice__c</code>	<p><b>For major cart items</b> The cart item's single unit price.</p> <p><b>For minor cart items</b> The cart item's single unit price multiplied by the cart item's original quantity.</p>

 **Note:** This field is set only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).

 **Note:** This method skips seller-based products and externally priced products, which already have a subtotal price specified.

SEE ALSO:

[Salesforce Help Create an Aggregated Product](#)

## `ccrz.ccLogicCartAddTo.concludeItemAdd`

Completes the For loop within the `ccrz.ccLogicCartAddTo.addCartLinesToItems` method by finalizing fields and relationships on the cart item.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> concludeItemAdd (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.validateItem` method.

This method requires the following key in the working map:

**ccrz.ccApiCart.CURR\_CARTITEM**

`ccrz__E_CartItem__c` that corresponds to the item currently being evaluated within the loop.

This method passes the working map through a chain of inner, non-overridable methods that:

- Set non-writable fields on the `ccrz__E_CartItem__c`
- Set the minor cart items for a major cart item that's a kit or bundle, based on the `compositeProductsS` related list returned from a `ccrz.ccApiProduct.fetch` query
- Prepares the cart item for upsert

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiCart.CARTITEMS\_TO\_UPSERT**

`List<ccrz__E_CartItem__c>` of cart items to upsert.

## ccrz.ccLogicCartAddTo.upsertCartItems

Executes the DML `upsert` operation for the created cart item records, which assigns IDs to each record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> upsertCartItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.validateItem` method.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CARTITEMS\_TO\_UPSERT**

`List<ccrz__E_CartItem__c>` of cart items to upsert.

**ccrz.ccApiCart.HAS\_LINE\_DATA**

Boolean

Value	Usage
<i>true</i>	The working map still contains line data that hasn't been evaluated.
<i>false</i>	The working map no longer contains any line data.

**ccrz.ccApiCart.RESPONSE**

Map<String, Object> that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The method upserted all cart items successfully.
<i>false</i>	The call encountered errors.

SEE ALSO:

[Upserting Records](#)

## ccrz.ccLogicCartAddTo.concludeAddTo

Returns final data keys, resets the cart's validation status, and optionally reprices and refetches cart records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



## Signature

```
global virtual Map<String, Object> concludeAddTo (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartAddTo.upsertCartItems` method.

This method requires the following key in the working map:

### **ccrz.ccApiCart.RESPONSE**

`Map<String, Object>` that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:


### **ccrz.ccApi.API\_SIZING**

`Map<String, Object>` that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the updated cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

### **ccrz.ccApiCart.ISREPRICE**

Boolean

Value	Usage
<i>true</i>	Call the <code>ccrz.ccApiCart.price</code> method to reprice the cart after the items are added.  <b>Note:</b> This operation can be expensive, depending on the complexity of your cart data, but ensures that prices are current and entitlement is correct.
<i>false</i> (default)	Don't reprice cart items.

### **ccrz.ccApiCart.PRODSADDEDTOCART**

`Map<ccrz__E_CartItem__c, String>` of SKUs or product IDs that were added to the cart.

## Outputs

This method returns the final `Map<String, Object>`, including the following keys:

### **ccrz.ccApiCart.CART\_OBJLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 Note:

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

This method also sets the following field on the corresponding `ccrz__E_Cart__c` record:

<code>ccrz__E_Cart__c</code> Field	Source for Value
<code>ccrz__ValidationStatus__c</code>	If the <code>ccrz.ccApiCart.PRODSADDEDTOCART</code> key contained data, which means that products were added to the cart, this field is reset.

## ccrz.ccLogicCartClone

Define the business logic for creating a copy of a cart or wish list.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiCart.cloneCart](#)

### Methods

#### IN THIS SECTION:

##### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

##### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

### fetchCarts

Calls the `ccrz.ccApiCart.fetch` method for returning data for the cart to clone.

### processData

Creates `ccrz__E_Cart__c`, `ccrz__E_CartItem__c`, and `ccrz__E_CartItemGroup__c` records that contain the data copied from the original cart.

### doDML

Executes the DML `insert` operation on the cloned `ccrz__E_Cart__c`, `ccrz__E_CartItem__c`, and `ccrz__E_CartItemGroup__c` records.

### prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## SEE ALSO:

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)

## ccrz.ccLogicCartClone.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.cloneCart` method.

## Logic Performed


This method invokes other methods of the `ccrz.ccLogicCartClone` class in the following order:

1. `processInput`

2. `fetchCarts`
3. `processData`
4. `doDML`
5. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## `ccrz.ccLogicCartClone.processInput`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartClone.process` method.


This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

**`ccrz.ccApiCart.CART_ID`**

String that specifies the Salesforce ID of the original cart to clone.

**`ccrz.ccApiCart.CART_IDLIST`**

`Set<String>` that can include the Salesforce ID of only one original cart to clone.


 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the original cart to clone.

**ccrz.ccApiCart.CART\_ENCIDLIST**

Set<String> that can include the encrypted ID of only one original cart to clone.

 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

 **Note:** This method checks for these keys in the following order of precedence:

1. ccrz.ccApiCart.CART\_ID
2. ccrz.ccApiCart.CART\_ENCID
3. ccrz.ccApiCart.CART\_IDLIST
4. ccrz.ccApiCart.CART\_ENCIDLIST

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The [ccrz.ccLogicCartClone.prepReturn method](#) removes this intermediate key from the working map.

## ccrz.ccLogicCartClone.fetchCarts

Calls the `ccrz.ccApiCart.fetch` method for returning data for the cart to clone.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchCarts (Map<String, Object>)
```

## Inputs (Required)

This method accepts the Map<String, Object> from the [ccrz.ccLogicCartClone.processInput](#) method.


This method requires one of the following keys to execute:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the original cart to clone.

**ccrz.ccApiCart.CART\_IDLIST**

Set<String> that can include the Salesforce ID of only one original cart to clone.


 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the original cart to clone.

**ccrz.ccApiCart.CART\_ENCIDLIST**

Set<String> that can include the encrypted ID of only one original cart to clone.

 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

 **Note:** This method checks for these keys in the following order of precedence:


1. ccrz.ccApiCart.CART\_ID
2. ccrz.ccApiCart.CART\_ENCID
3. ccrz.ccApiCart.CART\_IDLIST
4. ccrz.ccApiCart.CART\_ENCIDLIST

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, which contains one Map<String, Object> that represents the `ccrz__E_Cart__c` data to clone, which the `ccrz.ccApiCart.fetch` method returned.

 **Note:** By default, this logic service provider class clones only one cart record at one time. Even though this key specifies a list, the list contains only one cart.

**ccrz.ccApiCart.CARTITEMSBYID**

Map<String, List<ccrz\_\_E\_CartItem\_\_c>>, keyed by cart ID, for the fetched `ccrz__E_Cart__c` record.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCartClone.processData

Creates `ccrz__E_Cart__c`, `ccrz__E_CartItem__c`, and `ccrz__E_CartItemGroup__c` records that contain the data copied from the original cart.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartClone.fetchCarts` method.

This method requires the following keys in the working map:

### `ccrz.ccApiCart.CART_OBJLIST`

`List<Map<String, Object>>`, which contains one `Map<String, Object>` that represents the `ccrz__E_Cart__c` data to clone, which the `ccrz.ccApiCart.fetch` method returned.



**Note:** By default, this logic service provider class clones only one cart record at one time. Even though this key specifies a list, the list contains only one cart.

### `ccrz.ccApiCart.CARTITEMSBYID`

`Map<String, List<ccrz__E_CartItem__c>>`, keyed by cart ID, for the fetched `ccrz__E_Cart__c` record.

## Logic Performed

This method completes the following actions:

1. Creates a clone of the original `ccrz__E_Cart__c` record, and assigns a random integer for the clone's `ccrz__CartId__c`.
2. Sets the clone as the active cart, unless the original cart is a wish list.
3. Loops through the original cart's `ccrz__E_CartItem__c` and `ccrz__E_CartItemGroup__c` records. This method creates corresponding clones of the items and groups, and assigns a random integer for each clone's ID.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccLogicCartClone.CART_TO_INSERT`

`ccrz__E_Cart__c` that represents the cloned cart that this method created.

### `ccrz.ccLogicCartClone.CARTITEMS_TO_INSERT`

`List<ccrz__E_CartItem__c>` that contains clone's items.

### `ccrz.ccLogicCartClone.CHILD_PARENT`

`Map<String, Object>` that relates each minor `ccrz__E_CartItem__c` record's Salesforce ID (from the original cart) to the corresponding major `ccrz__E_CartItem__c` record's external ID (in the cloned cart).

**ccrz.ccLogicCartClone.GROUPS\_TO\_INSERT**

List<ccrz\_\_E\_CartItemGroup\_\_c> that contains cart item groups related to the cloned cart.

**ccrz.ccLogicCartClone.OLD\_CART**

ccrz\_\_E\_Cart\_\_c that represents the original cart.

**ccrz.ccLogicCartClone.OLD\_ITEM**

Map<String, Object> that relates each original ccrz\_\_E\_CartItem\_\_c record's Salesforce ID to its cloned ccrz\_\_E\_CartItem\_\_c record's external ID.

**ccrz.ccLogicCartClone.OLD\_GROUP**

Map<String, Object> that relates each original ccrz\_\_E\_CartItemGroup\_\_c record's Salesforce ID to its cloned ccrz\_\_E\_CartItemGroup\_\_c record's external ID.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartClone.doDML

Executes the DML `insert` operation on the cloned ccrz\_\_E\_Cart\_\_c, ccrz\_\_E\_CartItem\_\_c, and ccrz\_\_E\_CartItemGroup\_\_c records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> doDML (Map<String, Object>)
```

### Inputs (Required)

This method accepts the Map<String, Object> from the [ccrz.ccLogicCartClone.processData](#) method.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CARTITEMSBYID**

Map<String, List<ccrz\_\_E\_CartItem\_\_c>>, keyed by cart ID, for the original ccrz\_\_E\_Cart\_\_c record.



**ccrz.ccLogicCartClone.CART\_TO\_INSERT**

`ccrz__E_Cart__c` that represents the cloned cart to insert.

**ccrz.ccLogicCartClone.CARTITEMS\_TO\_INSERT**

List<`ccrz__E_CartItem__c`> that contains the cloned cart's items to insert.

**ccrz.ccLogicCartClone.OLD\_CART**

`ccrz__E_Cart__c` that represents the original cart.

**ccrz.ccLogicCartClone.OLD\_ITEM**

Map<String, Object> that relates each original `ccrz__E_CartItem__c` record's Salesforce ID to its cloned `ccrz__E_CartItem__c` record's external ID.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApi.SIZING**

Map<String, Object> that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```



**Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

**ccrz.ccLogicCartClone.CHILD\_PARENT**

Map<String, Object> that relates each minor `ccrz__E_CartItem__c` record's Salesforce ID (from the original cart) to the corresponding major `ccrz__E_CartItem__c` record's external ID (in the cloned cart).

**ccrz.ccLogicCartClone.GROUPS\_TO\_INSERT**

List<`ccrz__E_CartItemGroup__c`> that contains cart item groups related to the cloned cart.

**ccrz.ccLogicCartClone.OLD\_GROUP**

Map<String, Object> that relates each original `ccrz__E_CartItemGroup__c` record's Salesforce ID to its cloned `ccrz__E_CartItemGroup__c` record's external ID.

## Logic Performed

This method completes the following actions:


1. Inserts the cloned cart, cart item groups, and cart items.
2. Matches minor cart line items with its corresponding major cart line items.
3. Matches cart items with its corresponding cart item groups.
4. Makes the original cart inactive, unless the original cart was a wish list.
5. If the original input data specified `ccrz.ccApi.SZ_REFETCH => true`, calls the `ccrz.ccApiCart.fetch` method for including the cloned cart in the return data.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cloned cart that this method inserted.

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the cloned cart that this method inserted.

### `ccrz.ccApiCart.CART_OBJLIST`

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the cloned `ccrz__E_Cart__c` record, which the `ccrz.ccApiCart.fetch` method returned.



#### Note:

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[Inserting and Updating Records](#)

## `ccrz.ccLogicCartClone.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartClone.doDML` method.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicCartClone.CART_TO_INSERT`
- `ccrz.ccLogicCartClone.CARTITEMS_TO_INSERT`
- `ccrz.ccLogicCartClone.CHILD_PARENT`
- `ccrz.ccLogicCartClone.GROUPS_TO_INSERT`
- `ccrz.ccLogicCartClone.OLD_CART`
- `ccrz.ccLogicCartClone.OLD_GROUP`
- `ccrz.ccLogicCartClone.OLD_ITEM`

## ccrz.ccLogicCartCreate

Define the business logic for creating a cart record. By default, the methods of this class create only one cart record at a time. Even though the methods accept and return a list of cart records, the methods evaluate only the first element in the list.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiCart.create`

### Methods

#### IN THIS SECTION:

##### `process`

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

##### `processInput`

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

##### `processCartData`

Evaluates the input data for field values to set on the created cart, and specifies fallback values for any fields that aren't specified.

##### `createCarts`

Creates the cart sObject to insert.

##### `insertCarts`

Executes the DML `insert` operation on the cart record to create.

##### `createCartItems`

Creates the line data for the cart's items.

##### `insertCartItems`

Calls the `ccrz.ccApiCart.addTo` method for adding the specified cart items to the inserted cart.

##### `refetch`

Calls the `ccrz.ccApiCart.fetch` method for fetching the created cart record to include in the global API return data.

##### `prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## `ccrz.ccLogicCartCreate.process`

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.create` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartCreate` class in the following order:

1. `processInput`
2. `processCartData`
3. `createCarts`
4. `insertCarts`
5. `createCartItem`s
6. `insertCartItem`s
7. `refetch`
8. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartCreate.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartCreate.process` method.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### 'originalInputData'

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartCreate.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicCartCreate.processCartData

Evaluates the input data for field values to set on the created cart, and specifies fallback values for any fields that aren't specified.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processCartData (Map<String, Object>)
```

## Inputs (Required)


This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartCreate.processInput` method.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:


### **ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the [ccrz\\_\\_E\\_Cart\\_\\_c](#) record to create. This map can specify subkeys for field values to set on the cart record. For a list of fields, see the [ccrz\\_\\_E\\_Cart\\_\\_c](#) reference.

 **Note:** By default, this method creates only one cart record at a time. Even though this key specifies a list, the method evaluates only the first item in the list.

This method sets fallback values for the following fields if they're not specified in `ccrz.ccApiCart.CART_OBJLIST`:

Field	Fallback Value
activeCart	<i>true</i>
cartStatus	<i>Open</i>
currencyISOCode	<a href="#">ccrz.cc_CallContext.userCurrency</a>
name	<i>New Cart</i>
ownerId	<a href="#">ccrz.cc_CallContext.currUserId</a>
storefront	<a href="#">ccrz.cc_CallContext.storefront</a>


 **Note:** If you don't specify `ccrz.ccApiCart.CART_OBJLIST`, or if it contains an empty list, this method creates an empty cart with default values.

## Outputs

This method returns the working Map<String, Object>, and sets the following keys:

### 'cartItemsToCreate'

List<Map<String, Object>> where each Map<String, Object> describes data for a [ccrz\\_\\_E\\_CartItem\\_\\_c](#) associated with the cart.

 **Note:** The method returns this key when the `ccrz.ccApiCart.CART_OBJLIST` input key includes the `E_CartItemsS` subkey. The [ccrz.ccLogicCartCreate.createCartItems](#) method evaluates this key.

### **ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the [ccrz\\_\\_E\\_Cart\\_\\_c](#) record to create. If the `ccrz.ccApiCart.CART_OBJLIST` input was empty, this output includes data for a new empty cart with fallback values.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartCreate.createCarts

Creates the cart sObject to insert.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> createCarts (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartCreate.processCartData` method.

This method requires the following keys in the working map:

#### **ccrz.ccApiCart.CART\_OBJLIST**

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the `ccrz__E_Cart__c` record to create.

#### **ccrz.ccService.TRANSFORMINDEX**

`Map<String, Object>` that the `ccrz.ccServiceCart` data service provider generates for defining how the fields in a map convert to sObject data.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

#### **ccrz.ccApiCart.CART\_OBJLIST**

`List<ccrz__E_Cart__c>` that contains the cart sObject to insert.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[Working with Objects](#)

## ccrz.ccLogicCartCreate.insertCarts

Executes the DML `insert` operation on the cart record to create.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> insertCarts (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartCreate.createCarts` method.

This method requires the following key in the working map:

### `ccrz.ccApiCart.CART_OBJLIST`

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the `ccrz__E_Cart__c` record to create.

After the `insert` operation completes, this method also completes the following actions:


1. Calls the `ccrz.ccApiCart.setActive` method to make the inserted cart active, and deactivate the account's other open carts.
2. Calls the `ccrz.ccApiCart.fetch` method to query for the inserted cart's encrypted ID.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiCart.BYSTOREFRONT**

String that specifies the name of the storefront for the inserted cart.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the inserted cart.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the inserted cart.

**'newCartObjects'**

List<[ccrz\\_\\_E\\_Cart\\_\\_c](#)> that contains the inserted cart sObject.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartCreate.createCartItems

Creates the line data for the cart's items.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> createCartItems (Map<String, Object>)
```

### Inputs (Required)

This method accepts the Map<String, Object> from the [ccrz.ccLogicCartCreate.insertCarts](#) method.

This method requires the following keys in the working map:

**'cartItemsToCreate'**

List<Map<String, Object>> where each Map<String, Object> represents data for a [ccrz\\_\\_E\\_CartItem\\_\\_c](#) associated with the cart.



**Note:** The [ccrz.ccLogicCartCreate.processCartData](#) method created this key when the original input's [ccrz.ccApiCart.CART\\_OBJLIST](#) key included the [E\\_CartItemsS](#) subkey.

**'newCartObjects'**

List<[ccrz\\_\\_E\\_Cart\\_\\_c](#)> that contains the inserted cart sObject.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**'cartItemsToCreate'**

List<Map<String, Object>> where each Map<String, Object> describes data for a [ccrz\\_\\_E\\_CartItem\\_\\_c](#) associated with the cart. This method sets the following key values:

**ccrz.ccApiCart.LINE\_DATA\_QUANTITY**

Integer that specifies the quantity for the product to add to the cart.

**ccrz.ccApiCart.LINE\_DATA\_PRODUCT\_SFID**

Salesforce ID for the product to add to the cart.



**Note:** The [ccrz.ccLogicCartCreate.insertCartItems](#) method uses the content in 'cartItemsToCreate' as the [ccrz.ccApiCart.LINE\\_DATA](#) input for a [ccrz.ccApiCart.addTo](#) call.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartCreate.insertCartItems

Calls the [ccrz.ccApiCart.addTo](#) method for adding the specified cart items to the inserted cart.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> insertCartItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the Map<String, Object> from the [ccrz.ccLogicCartCreate.createCartItems](#) method.

This method requires the following keys in the working map:

**'cartItemsToCreate'**

List<Map<String, Object>> where each Map<String, Object> describes data for a [ccrz\\_\\_E\\_CartItem\\_\\_c](#) associated with the cart. This method uses this list for the `ccrz.ccApiCart.LINE_DATA` input for the [ccrz.ccApiCart.addTo](#) call.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the inserted cart to add items to.

## Outputs

This method returns the working Map<String, Object>, updated with the results of the [ccrz.ccApiCart.addTo](#) call.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccApiCart.addTo](#)

## ccrz.ccLogicCartCreate.refetch

Calls the `ccrz.ccApiCart.fetch` method for fetching the created cart record to include in the global API return data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)

This method accepts the Map<String, Object> from the [ccrz.ccLogicCartCreate.insertCartItems](#) method.

This method requires the following keys in the working map:

**ccrz.ccApi.SIZING**

Map<String, Object> that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```



**Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the created cart to query.

## Outputs

This method returns the working Map<String, Object>, updated with the results of the `ccrz.ccApiCart.fetch` call.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCartCreate.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working Map<String, Object> from the `ccrz.ccLogicCartCreate.refetch` method.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccService.TRANSFORMINDEX`
- `'newCartObjects'`
- `'originalInputData'`

## ccrz.ccLogicCartCreateCartItemGroup

Define the business logic for creating a shipping group of cart items for line-level independence (LLI) checkout.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiCart.createCartItemGroup](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [createItemGroups](#)

Evaluates the input fields and values and creates corresponding `ccrz__E_CartItemGroup__c` records. This method also wraps the DML `insert` statement for inserting the created records.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCartCreateCartItemGroup.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.createCartItemGroup` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartCreateCartItemGroup` class in the following order:

1. `processInput`
2. `createItemGroups`
3. `prepReturn`

### Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartCreateCartItemGroup.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartCreateCartItemGroup.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### 'ECartItemGroupsS'

`List<Map<String, Object>>`, where each `Map<String, Object>` represents the `ccrz__E_CartItemGroup__c` record to create. This map can specify subkeys for field values to set on the cart item group record. For a list of fields, see the `ccrz__E_CartItemGroup__c` reference.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### ccrz.ccLogic.ORIGINAL\_INPUT

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartCreateCartItemGroup.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicCartCreateCartItemGroup.createItemGroups

Evaluates the input fields and values and creates corresponding `ccrz__E_CartItemGroup__c` records. This method also wraps the DML `insert` statement for inserting the created records.

## Compatibility

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createItemGroups (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartCreateCartItemGroup.processInput` method.

This method requires the following key in the working map:

### 'ECartItemGroupsS'

`List<Map<String, Object>>`, where each `Map<String, Object>` represents the `ccrz__E_CartItemGroup__c` record to create. This map can specify subkeys for field values to set on the cart item group record. For a list of fields, see the `ccrz__E_CartItemGroup__c` reference.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### ccrz.ccApi.SIZING

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiStoredPayment.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```



**Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.


## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### ccrz.ccApi.SUCCESS


Boolean

Value	Usage
true	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the updated `ccrz__E_Cart__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

**ccrz.ccApiCart.ITEMGROUPID**

String that specifies the Salesforce ID of the created cart item group.

SEE ALSO:

[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)

[Inserting and Updating Records](#)

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

**ccrz.ccLogicCartCreateCartItemGroup.prepReturn**

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartCreateCartItemGroup.createItemGroups` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicCartCreateWL

Define the business logic for creating a wish list and wish list items based on a specific cart ID and its cart items.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiCart.createWishList`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [fetchCart](#)

Calls the `ccrz.ccApiCart.fetch` method for returning the cart to create a wish list from.

#### [createWishList](#)

Calls the `ccrz.ccApiWishList.create` method for creating the wish list record.

[refetch](#)

Calls the `ccrz.ccApiWishList.fetch` method for fetching the created wish list record to include in the global API return data.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## SEE ALSO:

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)

## ccrz.ccLogicCartCreateWL.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.createWishList` method.

### Logic Performed


This method invokes other methods of the `ccrz.ccLogicCartCreateWL` class in the following order:

1. `processInput`
2. `fetchCart`
3. `createWishList`
4. `refetch`

## 5. prepReturn

### Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartCreateWL.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartCreateWL.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

**ccrz.ccApiCart.CART\_ID**

The Salesforce ID of the cart to create a wish list from.

**ccrz.ccApiCart.CART\_ENCID**

The encrypted ID of the cart to create a wish list from.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartCreateWL.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicCartCreateWL.fetchCart

Calls the `ccrz.ccApiCart.fetch` method for returning the cart to create a wish list from.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchCart (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartCreateWL.processInput` method.

This method can accept either of the following parameters:

#### **ccrz.ccApiCart.CART\_ID**

The Salesforce ID of the cart to create a wish list from.

#### **ccrz.ccApiCart.CART\_ENCID**

The encrypted ID of the cart to create a wish list from.

### Logic Performed

This method completes the following actions:

1. Calls the `ccrz.ccApiCart.fetch` method using the `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID`.
2. Sets up data for creating a wish list from the fetched cart, including converting the cart's `ccrz__E_CartItem__c` records to corresponding wish list items.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiWishList.WISHLIST**

Map<String, Object> that describes the wish list to create from the fetched cart.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

**ccrz.ccLogicCartCreateWL.createWishList**

Calls the `ccrz.ccApiWishList.create` method for creating the wish list record.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> createWishList(Map<String, Object>)
```

**Inputs (Required)**

This method accepts the Map<String, Object> from the `ccrz.ccLogicCartCreateWL.fetchCart` method.

This method requires the following key in the working map:

**ccrz.ccApiWishList.WISHLIST**

Map<String, Object> that describes the wish list to create from the fetched cart.

**Logic Performed**

This method passes `ccrz.ccApiWishList.WISHLIST` as input data for the `ccrz.ccApiWishList.create` method.

**Outputs**

This method returns the working Map<String, Object>, which includes the output data from the `ccrz.ccApiWishList.create` method.

**ccrz.ccLogicCartCreateWL.refetch**

Calls the `ccrz.ccApiWishList.fetch` method for fetching the created wish list record to include in the global API return data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)


This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartCreateWL.createWishList` method.

This method requires the following key in the working map:

### **ccrz.ccApi.SIZING**

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiWishList.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

## Outputs

This method returns the working `Map<String, Object>`, which includes the output data from the `ccrz.ccApiWishList.fetch` method.

## ccrz.ccLogicCartCreateWL.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartCreateWL.refetch` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicCartInitItemGroups

Define the business logic for initializing shipping groups on a specific cart for line-level independence (LLI) checkout.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiCart.initItemGroups`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [initItemGroups](#)

Determines whether the cart already contains at least shipping group, and assigns cart items to the group. When no shipping group exists, this method creates a single default group with all cart items and the account's default shipping address. This method also wraps the DML `update` statement for assigning cart items.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCartInitItemGroups.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.initItemGroups` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartInitItemGroups` class in the following order:

1. `processInput`
2. `initItemGroups`
3. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## `ccrz.ccLogicCartInitItemGroups.processInput`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartInitItemGroups.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

### `ccrz.ccApiCart.CART_ID`

The Salesforce ID of the cart to initialize shipping groups for.

### `ccrz.ccApiCart.CART_ENCID`

The encrypted ID of the cart to initialize shipping groups for.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccLogic.ORIGINAL_INPUT`

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartInitItemGroups.prepReturn` method removes this intermediate key from the working map.

## `ccrz.ccLogicCartInitItemGroups.initItemGroups`

Determines whether the cart already contains at least shipping group, and assigns cart items to the group. When no shipping group exists, this method creates a single default group with all cart items and the account's default shipping address. This method also wraps the DML `update` statement for assigning cart items.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> initItemGroups (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartInitItemGroups.processInput` method.

This method can accept either of the following parameters:

### `ccrz.ccApiCart.CART_ID`

The Salesforce ID of the cart to initialize shipping groups for.

### `ccrz.ccApiCart.CART_ENCID`

The encrypted ID of the cart to initialize shipping groups for.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccApi.SIZING**

Map<String, Object> that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiStoredPayment.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

**Logic Performed**

This method completes the following actions:


1. Calls the `ccrz.ccApiCart.fetch` method and returns the specified cart.
2. If at least one shipping group exists on the cart, this method assigns any unassigned cart items to that group.
3. If no shipping group exists on the cart, this method completes the following actions:
  - Creates a single default shipping group for the cart.
  - Sets the group's shipping address to the account's default shipping address.
  - Adds all the cart items to the group.
4. Executes the DML `update` operation for assigning cart items to the group.
5. Sets the value of the cart's `ccrz__ShipStructure__c` field to `LLI`.

**Outputs**

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApi.SUCCESS**

Boolean


Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiCart.CART\_ENCID**

The encrypted ID of the updated cart.

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the updated `ccrz__E_Cart__c` record.

 **Note:** This key is returned only when the input map includes `ccrz.ccApi.SZ_REFETCH => true`.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)

[Inserting and Updating Records](#)

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

## ccrz.ccLogicCartInitItemGroups.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartInitItemGroups.initItemGroups` method.

### Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicCartGetActive

Define the business logic for resolving which cart is active for the current user, account (including effective accounts), storefront, and currency.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiCart.getActive](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [fetchActive](#)

Calls the `ccrz.ccApiCart.fetch` method for returning the active cart for a specified owner, storefront, currency, and effective account, if applicable. If the query can't find a match, this method also defines fallback queries for evaluating potentially active carts and selecting the best active cart candidate.

#### [createCart](#)

Calls the `ccrz.ccApiCart.create` method for creating a cart record that becomes the active cart when no other candidate active carts are available.

#### [updateActive](#)

Calls the `ccrz.ccApiCart.setActive` method for establishing a single active cart for a particular owner, and updating the owner's other open carts to inactive.

#### [refetch](#)

Calls the `ccrz.ccApiCart.fetch` method for fetching the updated active cart record to include in the global API return data.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCartGetActive.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.getActive` method.

### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartGetActive` class in the following order:

1. `processInput`
2. `fetchActive`
3. `createCart`
4. `updateActive`
5. `refetch`
6. `prepReturn`

### Outputs


This method creates a `Map<String, Object>` that contains a copy of the original input data. The method adds the following intermediate key to this map:

#### **ccrz.ccApiCart.RESPONSE**

`Map<String, Object>` that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.



 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartGetActive.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartGetActive.process` method.

This method requires the following key in the working map:

#### **ccrz.ccApiCart.RESPONSE**

`Map<String, Object>` that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

### Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

#### **ccrz.ccApiCart.BYEFFECTIVEACCOUNT**

Boolean

Value	Usage
<code>true</code>	Filter the cart query by the selected effective account. This method evaluates this key only when your storefront uses the <a href="#">Cart per Account mode for effective accounts</a> .
<code>false</code> (default)	Ignore effective accounts.

#### **ccrz.ccApiCart.BYOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

 **Note:**

- Sharing rules in your org can affect what data you can see when you filter by cart owner.
- If you don't include this key, this method falls back to `ccrz.cc_CallContext.currUserId`.

**ccrz.ccApiCart.BYSTOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to query.

**ccrz.ccApiCart.CART\_IDLIST**


Set<String> of Salesforce IDs of carts to query.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to query.


**ccrz.ccApiCart.CART\_ENCIDLIST**

Set<String> of encrypted cart IDs to query.

 **Note:** If you don't specify `ccrz.ccApiCart.CART_ID`, `ccrz.ccApiCart.CART_IDLIST`, `ccrz.ccApiCart.CART_ENCID`, or `ccrz.ccApiCart.CART_ENCIDLIST`, this method falls back to `ccrz.cc_CallContext.currCartId`.

**ccrz.ccApiCart.CURRCODE**

String that specifies an ISO 4217 currency code, such as *USD* or *JPY*.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userCurrency`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccLogicCartGetActive.CARTIDLIST**

Set<String> of Salesforce IDs of the carts to query.

**ccrz.ccLogicCartGetActive.CARTENCIDLIST**

Set<String> of encrypted IDs of the carts to query.

 **Note:**

- The `ccrz.ccLogicCartGetActive.CARTIDLIST` or `ccrz.ccLogicCartGetActive.CARTENCIDLIST` set is supposed to contain only one ID, but is established for the subsequent `ccrz.ccApiCart.fetch` query invoked from `ccrz.ccLogicCartGetActive.fetchActive`.
- Don't combine `ccrz.ccLogicCartGetActive.CARTIDLIST` and `ccrz.ccLogicCartGetActive.CARTENCIDLIST`.
- Always return one of these sets instead of passing through `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID`, respectively.

**ccrz.ccLogicCartGetActive.CARTID\_ACTIVE**

String that specifies the Salesforce ID of a candidate active cart. This method sets an initial null value, which subsequent methods update.

**ccrz.ccLogicCartGetActive.CARTLIST\_ACTIVE**

List<[ccrz\\_\\_E\\_Cart\\_\\_c](#)> of candidate active carts. This method instantiates an empty list, which subsequent methods update.

**ccrz.ccLogicCartGetActive.CARTOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

**ccrz.ccLogicCartGetActive.CURRENCYCODE**

String that specifies the currency for the cart query.

**ccrz.ccLogicCartGetActive.EFFECTIVEACCOUNT**

String that specifies the ID of the selected effective account for the cart query. This method sets this value from [ccrz.cc\\_CallContext.effAccountId](#).

**ccrz.ccLogicCartGetActive.STOREFRONT**

String that specifies the storefront for the cart query.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[Manage Carts for Effective Accounts](#)

## ccrz.ccLogicCartGetActive.fetchActive

Calls the [ccrz.ccApiCart.fetch](#) method for returning the active cart for a specified owner, storefront, currency, and effective account, if applicable. If the query can't find a match, this method also defines fallback queries for evaluating potentially active carts and selecting the best active cart candidate.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchActive (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the [ccrz.ccLogicCartGetActive.processInput](#) method.

This method executes only if the working map includes either of the following keys:

**ccrz.ccLogicCartGetActive.CARTIDLIST**

Set<String> of Salesforce IDs of the carts to query.

**ccrz.ccLogicCartGetActive.CARTENCIDLIST**

Set<String> of encrypted IDs of the carts to query.

 **Note:**

- The `ccrz.ccLogicCartGetActive.CARTIDLIST` or `ccrz.ccLogicCartGetActive.CARTENCIDLIST` set is supposed to contain only one ID, but is established for the subsequent `ccrz.ccApiCart.fetch` query invoked from `ccrz.ccLogicCartGetActive.fetchActive`.
- Don't combine `ccrz.ccLogicCartGetActive.CARTIDLIST` and `ccrz.ccLogicCartGetActive.CARTENCIDLIST`.

This method also requires the following keys:

**ccrz.ccApiCart.RESPONSE**

Map<String, Object> that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

**ccrz.ccLogicCartGetActive.CARTID\_ACTIVE**

String that specifies the Salesforce ID of a candidate active cart. This key's initial value is null.

**ccrz.ccLogicCartGetActive.CARTLIST\_ACTIVE**

List<`ccrz__E_Cart__c`> of candidate active carts. This list is initially empty.

**ccrz.ccLogicCartGetActive.CARTOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

**ccrz.ccLogicCartGetActive.CURRENCYCODE**

String that specifies the currency for the cart query.

**ccrz.ccLogicCartGetActive.STOREFRONT**

String that specifies the storefront for the cart query.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccLogicCartGetActive.EFFECTIVEACCOUNT**

String that specifies the ID of the selected effective account for the cart query. This method evaluates this key only when your storefront uses the [Cart per Account mode for effective accounts](#).

## Logic Performed

This method completes the following actions:


1. Applies the input keys to a `ccrz.ccApiCart.fetch` query, which returns the carts specified by `ccrz.ccLogicCartGetActive.CARTIDLIST` or `ccrz.ccLogicCartGetActive.CARTENCIDLIST`.
2. If the method doesn't match cart IDs, the method queries for carts where the `ccrz__ActiveCart__c` field is `TRUE`.
3. If the method still doesn't return any active carts, the method determines which of the returned carts is the best candidate to promote to active. By default, this method selects the cart with the most recent Last Modified Date.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, consider rolling back the API transaction to a previous savepoint.</p>

### `ccrz.ccLogicCartGetActive.CARTID_ACTIVE`

String that specifies the Salesforce ID of the cart that this method evaluated as active for the current call context.

### `ccrz.ccLogicCartGetActive.CARTLIST_ACTIVE`

`List<ccrz__E_Cart__c>` that includes the carts that this method fetched and evaluated as potentially active.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[Manage Carts for Effective Accounts](#)

## `ccrz.ccLogicCartGetActive.createCart`

Calls the `ccrz.ccApiCart.create` method for creating a cart record that becomes the active cart when no other candidate active carts are available.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8


## Signature

```
global virtual Map<String, Object> createCart (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartGetActive.fetchActive` method.

This method requires the following keys in the working map:

 **Note:** This method first checks the value of `ccrz.cc_CallContext.isGuest`, and executes only when the value is `false`.

### `ccrz.ccApiCart.RESPONSE`

`Map<String, Object>` that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

### `ccrz.ccApiCart.GETACTIVE_CREATE`

Boolean

Value	Usage
<code>true</code>	Create a cart record that becomes the active cart when no other candidate active carts are available.
<code>false</code> (default)	Don't create a cart record when no other candidate active carts are available.

### Important:

- This method executes only when this key's value is `true`.
- If you override this method in custom subscriber code, ensure that you never perform DML operations when this key's value is `true`.
- B2B Commerce for Visualforce sets this value to `true` for some internal use cases, such as in the `ccrz.ccApiCart.create` call invoked by the `ccrz.ccLogicCartAddTo.fetchCart` method.

### `ccrz.ccLogicCartGetActive.CARTID_ACTIVE`

String that specifies the Salesforce ID of the active cart that the `ccrz.ccLogicCartGetActive.fetchActive` method determined. This method executes only when this key's value is still null, indicating that the `ccrz.ccLogicCartGetActive.fetchActive` method couldn't find an active cart candidate.

### `ccrz.ccLogicCartGetActive.CARTOWNER`

String that specifies the ID of a specific user. The created cart's Owner field matches this user ID.

### `ccrz.ccLogicCartGetActive.CURRENCYCODE`

String that specifies the currency for the cart.

### `ccrz.ccLogicCartGetActive.STOREFRONT`

String that specifies the storefront for the cart.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### `ccrz.ccLogicCartGetActive.EFFECTIVEACCOUNT`


String that specifies the ID of the selected effective account for the cart.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccLogicCartGetActive.CARTID\_ACTIVE**

String that specifies the Salesforce ID of the created cart.

SEE ALSO:

[ccrz.ccApiCart.create](#)**ccrz.ccLogicCartGetActive.updateActive**

Calls the `ccrz.ccApiCart.setActive` method for establishing a single active cart for a particular owner, and updating the owner's other open carts to inactive.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8


**Signature**

```
global virtual Map<String, Object> updateActive (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartGetActive.createCart` method.

This method requires the following keys in the working map:

 **Note:** This method first checks the value of `ccrz.cc_CallContext.isGuest`, and executes only when the value is `false`.

**ccrz.ccApiCart.RESPONSE**

Map<String, Object> that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

**ccrz.ccApiCart.GETACTIVE\_UPDATE**

Boolean

Value	Usage
<code>true</code> (default)	Call the <code>ccrz.ccApiCart.setActive</code> method for establishing a single active cart and updating all other open carts to inactive.
<code>false</code>	Don't call the <code>ccrz.ccApiCart.setActive</code> method.

 **Important:**

- This method executes only when this key's value is `true`.
- If you override this method in custom subscriber code, ensure that you never perform DML operations when this key's value is `true`.

**ccrz.ccLogicCartGetActive.CARTID\_ACTIVE**

String that specifies the Salesforce ID of the cart to pass to the `ccrz.ccApiCart.setActive` method.

**ccrz.ccLogicCartGetActive.CARTOWNER**

String that specifies the ID of a specific user. The active cart's Owner field matches this user ID.

**ccrz.ccLogicCartGetActive.CURRENCYCODE**

String that specifies the currency for the cart.

**ccrz.ccLogicCartGetActive.STOREFRONT**

String that specifies the storefront for the cart.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccLogicCartGetActive.EFFECTIVEACCOUNT**

String that specifies the ID of the selected effective account for the cart.

## Outputs


This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.



Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccLogicCartGetActive.CARTLIST\_ACTIVE**

List<[ccrz\\_\\_E\\_Cart\\_\\_c](#)> that includes the cart that was made active.

SEE ALSO:

[ccrz.ccApiCart.setActive](#)

**ccrz.ccLogicCartGetActive.refetch**

Calls the [ccrz.ccApiCart.fetch](#) method for fetching the updated active cart record to include in the global API return data.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

**Inputs (Required)**

This method accepts the `Map<String, Object>` from the [ccrz.ccLogicCartGetActive.updateActive](#) method.

This method requires the following keys in the working map:

**ccrz.ccApi.SIZING**


`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

```

    }
}

```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

#### **ccrz.ccApiCart.RESPONSE**

Map<String, Object> that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

#### **ccrz.ccLogicCartGetActive.CARTID\_ACTIVE**

String that specifies the Salesforce ID of the active cart to query and include in the return data.

#### **ccrz.ccLogicCartGetActive.CARTOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

#### **ccrz.ccLogicCartGetActive.CURRENCYCODE**

String that specifies the currency for the cart query.

#### **ccrz.ccLogicCartGetActive.STOREFRONT**

String that specifies the storefront for the cart query.

This method applies these parameters to a `ccrz.ccApiCart.fetch` query for returning the active cart.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

#### **ccrz.ccLogicCartGetActive.EFFECTIVEACCOUNT**


String that specifies the ID of the selected effective account for the cart.

## Outputs

This method returns the working Map<String, Object>, which includes the `ccrz.ccApiCart.fetch` outputs, and sets the following keys:

#### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

#### **ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where each Map<String, Object> represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 **Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.

- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCartGetActive.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartGetActive.refetch` method.

This method requires the following key in the working map:

#### **ccrz.ccApiCart.RESPONSE**

`Map<String, Object>` that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

### Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApiCart.RESPONSE`
- `ccrz.ccLogicCartGetActive.CARTENCIDLIST`
- `ccrz.ccLogicCartGetActive.CARTID_ACTIVE`
- `ccrz.ccLogicCartGetActive.CARTIDLIST`

- `ccrz.ccLogicCartGetActive.CARTLIST_ACTIVE`
- `ccrz.ccLogicCartGetActive.CARTOWNER`
- `ccrz.ccLogicCartGetActive.CURRENCYCODE`
- `ccrz.ccLogicCartGetActive.EFFECTIVEACCOUNT`
- `ccrz.ccLogicCartGetActive.STOREFRONT`

## ccrz.ccLogicCartGetDeliveryDates

Define the business logic for returning the available delivery dates for one or more shipping addresses.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiCart.getDeliveryDates](#)

### Methods

#### IN THIS SECTION:

##### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

##### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

##### [getDeliveryDates](#)

Parses the input contact address fields, passes those values as the input parameters for the `ccrz.cc_api_shippingAndHandling.getShippingOptions` extension method, and returns that method's serialized data.

##### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCartGetDeliveryDates.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.getDeliveryDates` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartGetDeliveryDates` class in the following order:

1. `processInput`
2. `getDeliveryDates`
3. `prepReturn`

### Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartGetDeliveryDates.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartGetDeliveryDates.process` method.

This method checks that the input map specifies the following keys, and returns a `ccrz.ccApi.MissingInputException` if any key is missing:

### **ccrz.cc\_api\_DeliveryDate.CART\_ID**

String that specifies the ID of the cart record used for checkout.

### **ccrz.ccApiCart.SHIPPING\_ADDRLIST**

List<Map<String, Object>> where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c` record with shipping address details.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartGetDeliveryDates.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicCartGetDeliveryDates.getDeliveryDates

Parses the input contact address fields, passes those values as the input parameters for the

`ccrz.cc_api_ShippingAndHandling.getShippingOptions` extension method, and returns that method's serialized data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> getDeliveryDates (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartGetDeliveryDates.processInput` method.

This method requires the following keys in the working map:

### **ccrz.cc\_api\_DeliveryDate.CART\_ID**

String that specifies the ID of the cart record used for checkout.

### **ccrz.ccApiCart.SHIPPING\_ADDRLIST**

List<Map<String, Object>> where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c` record with shipping address details.

## Logic Performed


This method passes each address as the `ccrz.cc_api_DeliveryDate.SHIP_ADDRESS` input for the `ccrz.cc_api_DeliveryDate.getDeliveryDates` method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiCart.SHIPPING\_ADDRLIST**

Map<String, Object> where:

- The key is the contact address sfid.
- The object is a Map<String, Object> that the `ccrz.cc_api_DeliveryDate.getDeliveryDates` method returns.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz.cc\\_api\\_DeliveryDate.getDeliveryDates](#)

[Salesforce Help Specify Shipping Options for Standard Checkout](#)

**ccrz.ccLogicCartGetDeliveryDates.prepReturn**

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> from the `ccrz.ccLogicCartGetDeliveryDates.getDeliveryDates` method.

**Outputs**

This method removes the following intermediate key from the working map, and returns the final Map<String, Object>.

- `ccrz.ccLogic.ORIGINAL_INPUT`



## ccrz.ccLogicCartGetShippingOptions

Define the business logic for returning the available shipping options for one or more shipping addresses.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiCart.getShippingOptions](#)

### Methods

#### IN THIS SECTION:

##### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

##### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

##### [getShippingOptions](#)

Parses the input contact address fields, passes those values as the input parameters for the `ccrz.cc_api_shippingAndHandling.getShippingOptions` extension method, and returns that method's serialized data.

##### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### ccrz.ccLogicCartGetShippingOptions.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.getShippingOptions` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartGetShippingOptions` class in the following order:

1. `processInput`
2. `getShippingOptions`
3. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartGetShippingOptions.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartGetShippingOptions.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

### **ccrz.ccApiCart.SHIPPING\_ADDR**

`Map<String, Object>` that represents a `ccrz__E_ContactAddr__c` record with shipping address details.

### **ccrz.ccApiCart.SHIPPING\_ADDRLIST**

`List<Map<String, Object>>` where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c` record with shipping address details.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartGetShippingOptions.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicCartGetShippingOptions.getShippingOptions

Parses the input contact address fields, passes those values as the input parameters for the `ccrz.cc_api_ShippingAndHandling.getShippingOptions` extension method, and returns that method's serialized data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> getShippingOptions (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartGetShippingOptions.processInput` method.

This method requires either of the following keys in the working map:

### `ccrz.ccApiCart.SHIPPING_ADDR`

`Map<String, Object>` that represents a `ccrz__E_ContactAddr__c` record with shipping address details.

### `ccrz.ccApiCart.SHIPPING_ADDRLIST`

`List<Map<String, Object>>` where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c` record with shipping address details.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiCart.CART_ID`

String that specifies the ID of the cart used for checkout.

## Logic Performed

This method completes the following actions:


1. Reads the following fields from each contact address `Map<String, Object>` specified in the input data:
  - `countryISOCode`
  - `postalCode`
  - `sfid`
  - `stateISOCode`
2. Passes these values as the input parameters for the `ccrz.cc_api_ShippingAndHandling.getShippingOptions` method. If the input data included `ccrz.ccApiCart.CART_ID`, this method passes that ID as the value for the `ccrz.cc_api_ShippingAndHandling.getShippingOptions` method's `cartId` input.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

### `ccrz.ccApiCart.SHIPPING_OPTIONS`

The shipping options for the specified contact address or addresses. This return data can take the following types, depending on values specified in the input map:

Type	Input Map Criteria
<code>Map&lt;String, Object&gt;</code>	<code>ccrz.ccApiCart.SHIPPING_ADDR</code>
<code>List&lt;Map&lt;String, Object&gt;&gt;</code>	<code>ccrz.ccApiCart.SHIPPING_ADDRLIST</code>

For each `Map<String, Object>` returned:

- The key is the contact address `sfid`.
- The object is a serialized instance of `ccrz.cc_ctrl_hlpr_ShippingOption`, which the `ccrz.cc_api_ShippingAndHandling.getShippingOptions` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz.cc\\_api\\_ShippingAndHandling.getShippingOptions](#)

[ccrz.cc\\_ctrl\\_hlpr\\_ShippingOption](#)

[Salesforce Help Specify Shipping Options for Standard Checkout](#)

## `ccrz.ccLogicCartGetShippingOptions.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartGetShippingOptions.getShippingOptions` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicCartPlace

Define the business logic for converting a cart to an order.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**!** **Important:** In B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12):

- This logic service provider class includes methods that replace the `ccrz.cc_api_Cart.placeEOrder` public method, which is deprecated in this version. Pay close attention to the compatibility information listed for each inner method.
- As a result, this logic service provider executes differently compared to B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11) or older. The `ccrz.cc_api_Cart.placeEOrder` public method remains available for those older versions.
- To continue using `ccrz.cc_api_Cart.placeEOrder` logic for placing an order, set the Checkout storefront configuration module's Use Order Hook setting (`co.usehkororder`) to `TRUE`. This configuration specifies that methods of

the `ccrz.cc_hk_Order` extension point class execute and call `ccrz.cc_api_Cart.placeEOrder`. When this configuration is `FALSE`, the `ccrz.ccLogicCartPlace` logic service provider executes instead.

## Global API Caller

`ccrz.ccApiCart.place`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [checkCartForAsyncPlace](#)

Evaluates the cart, storefront configuration settings, and input keys to determine whether to place the order synchronously or asynchronously.

#### [retrieveCartHeader](#)

Calls the `ccrz.ccApiCart.fetch` method for querying the `ccrz__E_Cart__c` record to place an order from. This method can also evaluate the cart's `ccrz__ValidationStatus__c` before creating an order. Finally, this method also queries the cart's related `ccrz__E_CartCoupon__c` and `ccrz__E_ContactAddr__c` records, but doesn't query the cart's items.

#### [validateCartPricing](#)

Evaluates whether the cart's price data is current.

#### [processAddresses](#)

Copies the cart's billing and shipping addresses, so that the copies can apply to the order that this logic chain creates.

#### [createOrderHeader](#)

Creates the `ccrz__E_Order__c` record by copying all relevant field values from the source `ccrz__E_Cart__c` record. This method doesn't perform the DML `insert` operation.

#### [retrieveCartItems](#)

Calls the `ccrz.ccApiCart.fetchCartItems` method for querying the `ccrz__E_CartItem__c` records associated with the source `ccrz__E_Cart__c` record.

#### [processCartItems](#)

Separates the fetched `ccrz__E_CartItem__c` records into major and minor line items, and calls the `ccrz.ccLogicCartPlace.createOrderItem` method for creating corresponding `ccrz__E_OrderItem__c` records.

#### [createOrderItem](#)

Creates a `ccrz__E_OrderItem__c` record by copying all relevant field values from a source `ccrz__E_CartItem__c` record. This method also evaluates the source cart item's groups and subscriptions, and sorts the order item by whether it's a major or minor line item. The `ccrz.ccLogicCartPlace.processCartItems` method calls this method for each `ccrz__E_CartItem__c` associated with the source cart.

### [performDMLOnOrder](#)

Performs the DML `insert` operation on the order's billing and shipping addresses. Then, this method performs the DML `update` operation on the `ccrz__E_Order__c` record.

### [processOrderGroups](#)

Clones shipping addresses from source `ccrz__E_CartItemGroup__c` records for the corresponding `ccrz__E_OrderItemGroup__c` records. This method also inserts the created addresses and order item groups, and finally associates the order item groups with their corresponding order items.

### [performDMLOnOrderItems](#)

Calls inner methods for performing the DML `insert` operation for the order's major and minor line items, and then evaluates whether the cart has a coupon applied.

### [insertMajorItems](#)

Sets the `ccrz__Order__c` and `ccrz__RequestedDate__c` fields on each `ccrz__E_OrderItem__c` record that represents a major line item in the order. Then, the method performs the DML `insert` operation on the `ccrz__E_OrderItem__c` records.

### [insertMinorOrderItems](#)

Sets the `ccrz__Order__c`, `ccrz__RequestedDate__c`, and `ccrz__ParentOrderItem__c` fields on each `ccrz__E_OrderItem__c` record that represents a minor line item in the order. Then, the method performs the DML `insert` operation on the `ccrz__E_OrderItem__c` records.

### [processOrderTerms](#)

Creates and inserts `ccrz__E_OrderTerm__c` records for the order.

### [closeCart](#)

Updates fields on the source `ccrz__E_Cart__c` record to indicate that the cart is closed and a corresponding order was created.

### [createStoredPayments](#)

Evaluates whether to create a stored payment from the input payment data.

### [processSubscriptions](#)

Creates `ccrz__E_Subscription__c` records for any `ccrz__E_SubProdTerm__c` records associated with the order.

### [processTransactionPayments](#)

Evaluates whether to call the `ccrz.cc_hk_Order.createTransaction` extension point method, and then calls `ccrz.ccPaymentProcessor` methods for processing the payment.

### [changeOrderStatus](#)

Updates fields on the created `ccrz__E_Order__c` record to indicate that the order was placed.

### [processInvoices](#)

Calls the `ccrz.cc_hk_Invoice.onOrder` extension point method for creating invoice for the order.

### [refetch](#)

Call the `ccrz.ccApiCart.fetch` method for fetching the ordered cart.

### [placeOrder](#)

Passes the input data to the `ccrz.cc_api_Cart.placeEOrder` public method, which invokes methods of the `ccrz.cc_hk_Order` and `ccrz.cc_hk_Payment` extension point classes. Those methods return data to this method, including the created order and the transaction details.

### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.



[handleCartPlaceException](#)

Catches a `ccrz.ccLogicCartPlace.CartPlaceException` that occurs during the logic chain, and logs the exception message.

## ccrz.ccLogicCartPlace.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.place` method.

### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartPlace` class in the following order, depending on which version of the managed package you're using:

#### B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12)

1. `processInput`
2. `checkCartForAsyncPlace`



**Note:** If the `ccrz.ccApiCart.PLACE_ASYNC` key is `true` after this method completes, the `ccrz.ccApiCart.placeAsync` global method executes instead. The rest of the methods in this logic service provider class are skipped.

3. `retrieveCartHeader`
4. `validateCartPricing`
5. `processAddresses`
6. `createOrderHeader`

7. `retrieveCartItems`
8. `processCartItems`, which calls into `createOrderItem`
9. `performDMLOnOrder`
10. `processOrderGroups`
11. `performDMLOnOrderItems`, which calls into `insertMajorItems` and `insertMinorOrderItems`
12. `processOrderTerms`
13. `closeCart`
14. `createStoredPayments`
15. `processSubscriptions`
16. `processTransactionPayments`
17. `changeOrderStatus`
18. `processInvoices`
19. `refetch`
20. `prepReturn`

#### **B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11) or earlier**

1. `processInput`
2. `placeOrder`
3. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

This method handles exceptions differently depending on which version of the managed package you're using:

#### **B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12)**

This method rethrows any `ccrz.ccLogicCartPlace.CartPlaceException` instances that occur during the logic chain, which ends processing and rolls back transactions. The `ccrz.ccLogicCartPlace.handleCartPlaceException` method catches these exceptions.

#### **B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11) or earlier**

This method doesn't catch any exceptions that occur as the method executes.

SEE ALSO:

[ccrz.ccApiCart.placeAsync](#)

## `ccrz.ccLogicCartPlace.processInput`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlace.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

### **ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to convert.

### **ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to convert.

## Outputs (Any Version)

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartPlace.prepReturn` method removes this intermediate key from the working map.

## Outputs (API Version 12 or Later Only)

In B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12), this method also sets the following keys:

 **Important:** The following keys aren't set in any earlier version.

### **ccrz.ccApi.SUCCESS**

Boolean that's assigned an intermediate value of `false`. Subsequent methods in the logic chain update this value to `true` when the logic completes successfully.

### **ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

**ccrz.ccApiCart.ENFORCE\_PRICE\_VALIDATION\_ON\_PLACE**

Boolean that matches the value of the `ccrz.ccApiCart.ENFORCE_PRICE_VALIDATION_ON_PLACE` input. If this key wasn't specified in the input data, this method sets a fallback value:

Value	Usage
true	The Cart Scale storefront configuration module's Enable Cart Scale setting ( <code>cs.enable</code> ) is true.
false	The Cart Scale storefront configuration module's Enable Cart Scale setting ( <code>cs.enable</code> ) is false or null.

**ccrz.ccApiCart.PLACE\_ASYNC**

Boolean that matches the value of the `ccrz.ccApiCart.PLACE_ASYNC` input. If this key wasn't specified in the input data, this method sets a fallback value:

Value	Usage
true	The Checkout storefront configuration module's Process Orders Asynchronously setting ( <code>co.asyncorder</code> ) is true.
false	The Checkout storefront configuration module's Process Orders Asynchronously setting ( <code>co.asyncorder</code> ) is false or null.

**ccrz.ccLogicCartPlace.INVOICE\_ENABLED**

Boolean

Value	Usage
true	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is true.
false	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is false or null.

**ccrz.ccLogicCartPlace.checkCartForAsyncPlace**

Evaluates the cart, storefront configuration settings, and input keys to determine whether to place the order synchronously or asynchronously.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> checkCartForAsyncPlace (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.processInput` method completes.

This method requires either of the following keys in the working map:

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the cart to convert.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to convert.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### `ccrz.ccApiCart.FORCE_PLACE_SYNC`

Boolean

Value	Usage
<code>true</code>	Place the order synchronously, regardless of the value of the <code>ccrz.ccApiCart.PLACE_ASYNC</code> input.
<code>false</code> (default)	Default logic determines whether to place the order synchronously or asynchronously.

### `ccrz.ccApiCart.PLACE_ASYNC`

Boolean

Value	Usage
<code>true</code>	Place the order asynchronously by calling the <code>ccrz.ccApiCart.placeAsync</code> method.
<code>false</code> (default)	Place the order synchronously.

## Logic Performed

This method completes the following actions:

### 1. Evaluates whether any of the following criteria for *synchronous* cart pricing are satisfied:

- The `ccrz.ccApiCart.FORCE_PLACE_SYNC` input key is `true`.
- The buyer is a guest.
- The Cart Scale storefront configuration module's Enable Cart Scale setting (`cs.enable`) is `false` or null.

If any of these criteria are satisfied, this method sets `ccrz.ccApiCart.PLACE_ASYNC` to `false`, and the method exits. Otherwise, execution continues.

### 2. Evaluates whether the `ccrz.ccApiCart.PLACE_ASYNC` input key already specifies a value.

3. Completes the following actions when the Cart Scale storefront configuration module's Enable Cart Scale setting (`cs.enable`) is `true`:
  - Finds the value of the Cart Scale storefront configuration module's Asynchronous Processing Threshold setting (`cs.asyncthreshold`).
  - Passes `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID` as input for the `ccrz.ccApiCart.fetch` method.
  - Finds the value of the fetched `ccrz__E_Cart__c` record's `ccrz__NumberOfAllCartItems__c` field.
  - Compares the total number of cart items with the Asynchronous Processing Threshold value. When the number of cart items exceeds the processing threshold, the order is placed asynchronously.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCart.PLACE_ASYNC`

Boolean

Value	Usage
<code>true</code>	The criteria determined that the order is placed asynchronously. This logic service provider skips the remaining methods in its logic chain, and calls <code>ccrz.ccApiCart.placeAsync</code> instead.
<code>false</code>	The criteria determined that the order is placed synchronously. This logic service provider's execution continues.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)  
[ccrz.ccApiCart.fetch](#)

## `ccrz.ccLogicCartPlace.retrieveCartHeader`

Calls the `ccrz.ccApiCart.fetch` method for querying the `ccrz__E_Cart__c` record to place an order from. This method can also evaluate the cart's `ccrz__ValidationStatus__c` before creating an order. Finally, this method also queries the cart's related `ccrz__E_CartCoupon__c` and `ccrz__E_ContactAddr__c` records, but doesn't query the cart's items.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> retrieveCartHeader (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.checkCartForAsyncPlace` method completes.

This method requires either of the following keys in the working map:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to convert.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to convert.

 **Note:** If neither key is specified, this method returns a `ccrz.ccLogicCartPlace.CartPlaceException`.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiCart.SKIP\_VALID\_CHECK**

Boolean

Value	Usage
<i>true</i>	During this method's execution, ignore the cart's <code>ccrz__ValidationStatus__c</code> field and attempt to create an order from the cart.
<i>false</i> (default)	During this method's execution, verify that the cart's <code>ccrz__ValidationStatus__c</code> field is set before creating an order for this cart. If the cart isn't valid, this method returns a <code>ccrz.cc_api_Cart.CartValidationException</code> .

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiCart.fetch` outputs, and sets the following intermediate keys:

**ccrz.ccApiAddress.ADDRESSLIST**

`List<ccrz__E_ContactAddr__c>` of the cart's shipping or billing addresses. For more information about fetching address data with cart data, see `ccrz.ccApiCart.fetch`.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the fetched cart.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the fetched cart.

**ccrz.ccApiCart.HAS\_COUPON\_CODE**

Boolean

Value	Usage
<i>true</i>	The cart has a coupon applied.
<i>false</i>	The cart doesn't have any coupon applied.

**ccrz.ccLogicCartPlace.CART\_HEADER**

`ccrz__E_Cart__c` record that represents the cart to convert to an order.

**ccrz.ccLogicCartPlace.COUPON\_TO\_UPDATE**

List<`ccrz__E_CartCoupon__c`> that contains a coupon associated with the cart.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCartPlace.validateCartPricing

Evaluates whether the cart's price data is current.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> validateCartPricing(Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.retrieveCartHeader` method completes.

This method requires the following keys in the working map:

**ccrz.ccLogicCartPlace.CART\_HEADER**


`ccrz__E_Cart__c` record that represents the cart to convert to an order.

### Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccApiCart.ENFORCE\_PRICE\_VALIDATION\_ON\_PLACE**

Boolean

Value	Usage
<code>true</code>	Validate that the cart's price data is current before starting the order placement process.  <b>Note:</b> When you enable Cart Scale, this input is <code>true</code> by default.
<code>false</code> (default)	Don't validate that the cart's price data is current before starting order placement.



## Logic Performed

This method completes the following actions:

1. Checks whether the input data includes the `ccrz.ccApiCart.ENFORCE_PRICE_VALIDATION_ON_PLACE` key with a value of `true`:
  - If so, this method continues.
  - If the input's value is `false` or not specified, this method exits.
2. Evaluates the cart's `ccrz__RepricedDate__c` field.
  - If the cart's price data is current, the logic chain continues.
  - If the cart's price data isn't current, this method returns a `ccrz.cc_api_Cart.CartValidationException`.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

## ccrz.ccLogicCartPlace.processAddresses

Copies the cart's billing and shipping addresses, so that the copies can apply to the order that this logic chain creates.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processAddresses (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.validateCartPricing` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiAddress.ADDRESSLIST**

`List<ccrz__E_ContactAddr__c>` of the cart's shipping or billing addresses.

### **ccrz.ccLogicCartPlace.CART\_HEADER**

`ccrz__E_Cart__c` record that represents the cart to convert to an order.

## Logic Performed

This method completes the following actions:

1. Parses `ccrz.ccApiAddress.ADDRESSLIST` for the cart's `ccrz__BillTo__c` billing address, and clones the corresponding `ccrz__E_ContactAddr__c` record.
2. Parses `ccrz.ccApiAddress.ADDRESSLIST` for the cart's `ccrz__ShipTo__c` shipping address, and clones the corresponding `ccrz__E_ContactAddr__c` record.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccLogicCartPlace.ORDER_BILLING_ADDRESS`

`ccrz__E_ContactAddr__c` record cloned from the cart's billing address.

### `ccrz.ccLogicCartPlace.ORDER_SHIPPING_ADDRESS`

`ccrz__E_ContactAddr__c` record cloned from the cart's shipping address.

This method also removes the following intermediate key from the working map:

- `ccrz.ccApiAddress.ADDRESSLIST`

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

## `ccrz.ccLogicCartPlace.createOrderHeader`

Creates the `ccrz__E_Order__c` record by copying all relevant field values from the source `ccrz__E_Cart__c` record. This method doesn't perform the DML `insert` operation.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> createOrderHeader (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.processAddresses` method completes.

This method requires the following key in the working map:

**ccrz.ccLogicCartPlace.CART\_HEADER**

`ccrz__E_Cart__c` record that represents the cart to convert to an order.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccLogicCartPlace.ORDER\_HEADER**

`ccrz__E_Order__c` record with relevant fields copied from the source `ccrz__E_Cart__c` record.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_Order\\_\\_c](#)

## ccrz.ccLogicCartPlace.retrieveCartItems

Calls the `ccrz.ccApiCart.fetchCartItems` method for querying the `ccrz__E_CartItem__c` records associated with the source `ccrz__E_Cart__c` record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> retrieveCartItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.createOrderHeader` method completes.

This method requires the following key in the working map:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to query items for.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccLogicCartPlace.CART\_ITEM\_FETCH\_RESPONSE**

Map<String, Object> that contains the `ccrz.ccApiCart.fetchCartItems` output, including the fetched `ccrz__E_CartItem__c` records.

SEE ALSO:

[ccrz.ccApiCart.fetchCartItems](#)

**ccrz.ccLogicCartPlace.processCartItems**

Separates the fetched `ccrz__E_CartItem__c` records into major and minor line items, and calls the `ccrz.ccLogicCartPlace.createOrderItem` method for creating corresponding `ccrz__E_OrderItem__c` records.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

**Signature**

```
global virtual Map<String, Object> processCartItems (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> after the `ccrz.ccLogicCartPlace.retrieveCartItems` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the source cart.

**ccrz.ccLogicCartPlace.CART\_HEADER**

`ccrz__E_Cart__c` record that represents the source cart.

**ccrz.ccLogicCartPlace.CART\_ITEM\_FETCH\_RESPONSE**

Map<String, Object> that contains the `ccrz.ccApiCart.fetchCartItems` output, including the `ccrz__E_CartItem__c` records associated with the source cart.



**Note:** If this key doesn't reference any cart items, this method returns a `ccrz.ccLogicCartPlace.CartPlaceException`.

**ccrz.ccLogicCartPlace.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create items for.

**Inputs (Optional)**

If the working map includes the following key, this method evaluates it:

**ccrz.ccLogicCartPlace.INVOICE\_ENABLED**

Boolean

Value	Usage
<code>true</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>true</code> .
<code>false</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>false</code> or null.

**Logic Performed**

This method completes the following actions:

1. Parses `ccrz.ccLogicCartPlace.CART_ITEM_FETCH_RESPONSE` for `ccrz__E_CartItem__c` records and sorts the items into separate maps of major and minor line items.
2. Checks for any `ccrz__E_CartItemGroup__c` records from the source cart. If cart item groups exist, this method sets up corresponding `ccrz__E_OrderItemGroup__c` records, and builds maps that relate each group's items and groups for subsequent processing:
  - One map, `'cartGroupToOrderItemGroup'`, that maps *every* source `ccrz__E_CartItemGroup__c`—even groups that are empty and don't reference any individual `ccrz__E_CartItem__c` records—to its corresponding `ccrz__E_OrderItemGroup__c`.
  - A second map, `'orderIGroupsToAdd'`, that maps only `ccrz__E_CartItemGroup__c` records that aren't empty—that reference individual `ccrz__E_CartItem__c` records—with their corresponding `ccrz__E_OrderItemGroup__c` records. Later, the `ccrz.ccLogicCartPlace.processOrderGroups` method inserts only the `ccrz__E_OrderItemGroup__c` records that aren't empty.
  - A third map, `'cartGroupToOrderItems'`, that's initially empty.
  - Maps for managing the cart item group shipping addresses.
3. Calls the `ccrz.ccLogicCartPlace.createOrderItem` inner method for each `ccrz__E_CartItem__c`, which completes the following actions:
  - Creates a corresponding `ccrz__E_OrderItem__c`.
  - Associates the `ccrz__E_CartItem__c` with the appropriate `ccrz__E_OrderItemGroup__c` in `'orderIGroupsToAdd'`.
  - Associates the source `ccrz__E_CartItemGroup__c` with the list of created `ccrz__E_OrderItem__c` records in `'cartGroupToOrderItems'`.

**Outputs**

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccLogicCartPlace.MAJOR\_ORDER\_ITEMS**

`Map<String, ccrz__E_OrderItem__c>`, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is the `ccrz__E_OrderItem__c` to create for the corresponding major line item in the order.

**ccrz.ccLogicCartPlace.MINOR\_ORDER\_ITEMS**

`Map<String, List<ccrz__E_OrderItem__c>>`, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is a `List<ccrz__E_OrderItem__c>` to create as minor line items for the `ccrz__E_OrderItem__c` that corresponds to the major line item from the cart.

#### **ccrz.cc\_hk\_Invoice.PARAM\_CART\_ITEMS**

`List<ccrz__E_CartItem__c>` of the original cart items included in the `ccrz.ccLogicCartPlace.CART_ITEM_FETCH_RESPONSE` input.

 **Note:** This output is returned only when the input data specifies `ccrz.ccLogicCartPlace.INVOICE_ENABLED => true`.

#### **'addressToCartGroupMap'**

`Map<ID, ID>`, where:

- Each key is the Salesforce ID of the `ccrz__E_ContactAddr__c` record that the `ccrz__E_CartItemGroup__c` record's `ccrz__ShipTo__c` field references.
- Each value is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.

#### **'cartGroupAddressMap'**

`Map<ID, ID>`, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the Salesforce ID of the `ccrz__E_ContactAddr__c` record that the `ccrz__E_CartItemGroup__c` record's `ccrz__ShipTo__c` field references.

#### **'cartGroupToOrderItemGroup'**

`Map<ID, ccrz__E_OrderItemGroup__c>` that maps *every* source `ccrz__E_CartItemGroup__c`—even groups that are empty and don't reference any individual `ccrz__E_CartItem__c` records—to its corresponding `ccrz__E_OrderItemGroup__c`:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the corresponding `ccrz__E_OrderItemGroup__c` record that the method created.

#### **'cartGroupToOrderItems'**

`Map<ID, List<ccrz__E_OrderItem__c>>`, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is a `List<ccrz__E_OrderItem__c>` of the order items that the `ccrz.ccLogicCartPlace.createOrderItem` created and included in the group.

#### **'orderIGroupsToAdd'**

`Map<ID, ccrz__E_OrderItemGroup__c>` that maps only `ccrz__E_CartItemGroup__c` records that aren't empty—that reference individual `ccrz__E_CartItem__c` records—with their corresponding `ccrz__E_OrderItemGroup__c` records:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.

- Each value is the corresponding `ccrz__E_OrderItemGroup__c` record that the method created.

## SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)[ccrz\\_\\_E\\_OrderItemGroup\\_\\_c](#)[ccrz.ccLogicCartPlace.createOrderItem](#)

## ccrz.ccLogicCartPlace.createOrderItem

Creates a `ccrz__E_OrderItem__c` record by copying all relevant field values from a source `ccrz__E_CartItem__c` record. This method also evaluates the source cart item's groups and subscriptions, and sorts the order item by whether it's a major or minor line item. The `ccrz.ccLogicCartPlace.processCartItems` method calls this method for each `ccrz__E_CartItem__c` associated with the source cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> createOrderItem(Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlace.processCartItems` method.

This method requires the following keys in the working map:

**'cartItemToCreate'**

`ccrz__E_CartItem__c` record to create an `ccrz__E_OrderItem__c` from.

**'hasCurrencyIsoCode'**

Boolean

Value	Usage
<code>true</code>	The org supports multiple currencies, which indicates that the created <code>ccrz__E_OrderItem__c</code> must specify a value for the <code>currencyIsoCode</code> field.
<code>false</code> (default)	The storefront supports only one currency.

**'majorOrderItemMap'**

Map<String, ccrz\_\_E\_OrderItem\_\_c>, where:

- Each key is the Salesforce ID of a source ccrz\_\_E\_CartItem\_\_c record that represents a major line item in the cart.
- Each value is the ccrz\_\_E\_OrderItem\_\_c to create for the corresponding major line item in the order.

**'minorOrderItemMap'**

Map<String, List<ccrz\_\_E\_OrderItem\_\_c>>, where:

- Each key is the Salesforce ID of a source ccrz\_\_E\_CartItem\_\_c record that represents a major line item in the cart.
- Each value is a List<ccrz\_\_E\_OrderItem\_\_c> to create as minor line items for the ccrz\_\_E\_OrderItem\_\_c that corresponds to the major line item from the cart.

**'workingMap'**

Map<String, Object> that was passed to the ccrz.ccLogicCartPlace.processCartItems method and then to this method.

## Logic Performed

This method completes the following actions:

1. Creates an ccrz\_\_E\_OrderItem\_\_c from the source ccrz\_\_E\_CartItem\_\_c record.
2. Evaluates whether the source ccrz\_\_E\_CartItem\_\_c has an associated ccrz\_\_E\_SubProdTerm\_\_c.
3. Evaluates the order item's ccrz\_\_OrderLineType\_\_c and sorts the item into an appropriate map:
  - If the order item is a major line item or coupon, the method adds the order item to the map of major items.
  - Otherwise, the method adds the order item to the map of minor items.
4. Evaluates whether the cart item belonged to any ccrz\_\_E\_CartItemGroup\_\_c records, and processes the group information:
  - Associates the ccrz\_\_E\_CartItem\_\_c with the appropriate ccrz\_\_E\_OrderItemGroup\_\_c in the 'orderIGroupsToAdd' key in the working map.
  - Associates the source ccrz\_\_E\_CartItemGroup\_\_c with the list of created ccrz\_\_E\_OrderItem\_\_c records in the 'cartGroupToOrderItems' key in the working map.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**'hasSubProdTerm'**

Boolean

Value	Usage
true	The source ccrz__E_CartItem__c had an associated ccrz__E_SubProdTerm__c record.



Value	Usage
false	The source <code>ccrz__E_CartItem__c</code> didn't have subscription data.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[ccrz.ccLogicCartPlace.processCartItems](#)

## ccrz.ccLogicCartPlace.performDMLOnOrder

Performs the DML `insert` operation on the order's billing and shipping addresses. Then, this method performs the DML `update` operation on the `ccrz__E_Order__c` record.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> performDMLOnOrder (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.processCartItems` method completes.

This method requires the following keys in the working map:

**ccrz.ccLogicCartPlace.ORDER\_HEADER**


`ccrz__E_Order__c` record that represents the order to insert.

**'orderBillingAddress'**

`ccrz__E_ContactAddr__c` record cloned from the order's billing address to insert.

**'orderShippingAddress'**

`ccrz__E_ContactAddr__c` record cloned from the order's shipping address to insert.

 **Note:** The `ccrz.ccLogicCartPlace.processAddresses` method cloned the cart's billing and shipping addresses to create these `ccrz__E_ContactAddr__c` records.

### Logic Performed

This method completes the following actions:

1. Performs the DML `insert` operation on the `ccrz__E_ContactAddr__c` records for the order's billing and shipping addresses

2. Associates the inserted addresses with the `ccrz__E_Order__c` record.
3. Performs the DML `insert` operation on the `ccrz__E_Order__c` record.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCart.ORDER_ID`

String that specifies the Salesforce ID of the inserted `ccrz__E_Order__c` record.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz.ccLogicCartPlace.processAddresses](#)

[Inserting and Updating Records](#)

## `ccrz.ccLogicCartPlace.processOrderGroups`

Clones shipping addresses from source `ccrz__E_CartItemGroup__c` records for the corresponding `ccrz__E_OrderItemGroup__c` records. This method also inserts the created addresses and order item groups, and finally associates the order item groups with their corresponding order items.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processOrderGroups (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.performDMLOnOrder` method completes.

This method requires the following keys in the working map:

### `ccrz.ccLogicCartPlace.CART_HEADER`

`ccrz__E_Cart__c` record that represents the source cart.

### `ccrz.ccLogicCartPlace.ORDER_HEADER`

`ccrz__E_Order__c` record that represents the order to create.

### 'addressToCartGroupMap'

`Map<ID, ID>`, where:

- Each key is the Salesforce ID of the `ccrz__E_ContactAddr__c` record that the `ccrz__E_CartItemGroup__c` record's `ccrz__ShipTo__c` field references.
- Each value is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.

#### 'cartGroupAddressMap'

Map<ID, ID>, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the Salesforce ID of the `ccrz__E_ContactAddr__c` record that the `ccrz__E_CartItemGroup__c` record's `ccrz__ShipTo__c` field references.

#### 'cartGroupToOrderItemGroup'

Map<ID, `ccrz__E_OrderItemGroup__c`> that maps *every* source `ccrz__E_CartItemGroup__c`—even groups that are empty and don't reference any individual `ccrz__E_CartItem__c` records—to its corresponding `ccrz__E_OrderItemGroup__c`:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the corresponding `ccrz__E_OrderItemGroup__c` record.

#### 'cartGroupToOrderItems'

Map<ID, List<`ccrz__E_OrderItem__c`>>, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is a List<`ccrz__E_OrderItem__c`> of the order items that the `ccrz.ccLogicCartPlace.createOrderItem` created and included in the group.

#### 'orderIGroupsToAdd'

Map<ID, `ccrz__E_OrderItemGroup__c`> that maps only `ccrz__E_CartItemGroup__c` records that aren't empty—that reference individual `ccrz__E_CartItem__c` records—with their corresponding `ccrz__E_OrderItemGroup__c` records:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the corresponding `ccrz__E_OrderItemGroup__c` record.

## Logic Performed

This method completes the following actions:

1. Calls the `ccrz.ccApiAddress.fetch` method for querying the `ccrz__E_ContactAddr__c` records of the shipping addresses associated with each `ccrz__E_CartItemGroup__c`.
2. Calls the `ccrz.ccApiAddress.createAddress` method for cloning the fetched shipping addresses.
3. Assigns the appropriate shipping address to each corresponding `ccrz__E_OrderItemGroup__c`.
4. Performs the DML `insert` operation for each `ccrz__E_OrderItemGroup__c` record in 'orderIGroupsToAdd'. These groups reference the individual `ccrz__E_OrderItem__c` records to create.
5. Sets the `ccrz__OrderItemGroup__c` field on each `ccrz__E_OrderItem__c`.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)  
[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)  
[ccrz\\_\\_E\\_OrderItemGroup\\_\\_c](#)  
[ccrz.ccApiAddress.createAddress](#)  
[ccrz.ccApiAddress.fetch](#)  
[Inserting and Updating Records](#)

## ccrz.ccLogicCartPlace.performDMLOnOrderItems

Calls inner methods for performing the DML `insert` operation for the order's major and minor line items, and then evaluates whether the cart has a coupon applied.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> performDMLOnOrderItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.processOrderGroups` method completes.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:


**ccrz.ccApiCart.HAS\_COUPON\_CODE**

Boolean

Value	Usage
<i>true</i>	The cart has a coupon applied.
<i>false</i>	The cart doesn't have any coupon applied.

**ccrz.ccLogicCartPlace.COUPON\_TO\_UPDATE**

List<[ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)> that contains a coupon associated with the cart.

 **Note:** Although this input is a list, this method evaluates only the first record in the list.

## Logic Performed

This method completes the following actions:

1. Calls the [ccrz.ccLogicCartPlace.insertMajorItems](#) inner method.
2. Calls the [ccrz.ccLogicCartPlace.insertMinorOrderItems](#) inner method.
3. Evaluates the [ccrz.ccApiCart.HAS\\_COUPON\\_CODE](#) input, and calls the [ccrz.ccApiCoupon.updateOnUse](#) method for updating the coupon's usage.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiCoupon.COUPONS\_UPDATED**

List<[ccrz\\_\\_E\\_Coupon\\_\\_c](#)> of the updated coupon records that the [ccrz.ccApiCoupon.updateOnUse](#) method returned.

SEE ALSO:

[ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)

[ccrz.ccApiCoupon.updateOnUse](#)

## ccrz.ccLogicCartPlace.insertMajorItems

Sets the [ccrz\\_\\_Order\\_\\_c](#) and [ccrz\\_\\_RequestedDate\\_\\_c](#) fields on each [ccrz\\_\\_E\\_OrderItem\\_\\_c](#) record that represents a major line item in the order. Then, the method performs the DML `insert` operation on the [ccrz\\_\\_E\\_OrderItem\\_\\_c](#) records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> insertMajorItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the [ccrz.ccLogicCartPlace.performDMLOnOrderItems](#) method.

This method requires the following keys in the working map:

**ccrz.ccLogicCartPlace.MAJOR\_ORDER\_ITEMS**

Map<String, ccrz\_\_E\_OrderItem\_\_c>, where:

- Each key is the Salesforce ID of a source ccrz\_\_E\_CartItem\_\_c record that represents a major line item in the cart.
- Each value is the ccrz\_\_E\_OrderItem\_\_c to create for the corresponding major line item in the order.

**ccrz.ccLogicCartPlace.ORDER\_HEADER**

ccrz\_\_E\_Order\_\_c record that represents the order to create.

## Outputs

This method returns the working Map<String, Object>, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartPlace.insertMinorOrderItems

Sets the ccrz\_\_Order\_\_c, ccrz\_\_RequestedDate\_\_c, and ccrz\_\_ParentOrderItem\_\_c fields on each ccrz\_\_E\_OrderItem\_\_c record that represents a minor line item in the order. Then, the method performs the DML `insert` operation on the ccrz\_\_E\_OrderItem\_\_c records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> insertMinorOrderItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working Map<String, Object> from the [ccrz.ccLogicCartPlace.performDMLOnOrderItems](#) method.

This method requires the following keys in the working map:

**ccrz.ccLogicCartPlace.MAJOR\_ORDER\_ITEMS**

Map<String, ccrz\_\_E\_OrderItem\_\_c>, where:

- Each key is the Salesforce ID of a source ccrz\_\_E\_CartItem\_\_c record that represents a major line item in the cart.
- Each value is the ccrz\_\_E\_OrderItem\_\_c to create for the corresponding major line item in the order.

**ccrz.ccLogicCartPlace.MINOR\_ORDER\_ITEMS**

Map<String, List<ccrz\_\_E\_OrderItem\_\_c>>, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is a `List<ccrz__E_OrderItem__c>` to create as minor line items for the `ccrz__E_OrderItem__c` that corresponds to the major line item from the cart.

#### **ccrz.ccLogicCartPlace.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

#### **ccrz.ccLogicCartPlace.INVOICE\_ENABLED**

Boolean

Value	Usage
<code>true</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>true</code> .
<code>false</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>false</code> or null.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

#### **ccrz.cc\_hk\_Invoice.PARAM\_ORDER\_CHILD\_ITEMS**

`List<ccrz__E_OrderItem__c>` of the inserted minor line items.



**Note:** This output is returned only when the input data specifies `ccrz.ccLogicCartPlace.INVOICE_ENABLED => true`.

SEE ALSO:

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartPlace.processOrderTerms

Creates and inserts `ccrz__E_OrderTerm__c` records for the order.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processOrderTerms (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.performDMLOnOrderItems` method completes.

This method requires the following keys in the working map:

### `ccrz.ccLogicCartPlace.CART_HEADER`

`ccrz__E_Cart__c` record that represents the source cart.

### `ccrz.ccLogicCartPlace.ORDER_HEADER`

`ccrz__E_Order__c` record that represents the order to create.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccLogicCartPlace.INVOICE_ENABLED`

Boolean

Value	Usage
<i>true</i>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>true</code> .
<i>false</i>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>false</code> or null.

## Logic Performed

This method completes the following actions:

1. Queries the `ccrz__E_Term__c` records for the storefront.
2. Creates `ccrz__E_OrderTerm__c` records for the order, based on the following criteria:
  - When the source `ccrz__E_Cart__c` record's `ccrz__ShipMethod__c` field isn't null, this method creates an `ccrz__E_OrderTerm__c` that references a `ccrz__E_Term__c` record where `ccrz__Type__c` is Shipping.
  - When the source `ccrz__E_Cart__c` record's `ccrz__TotalAmount__c` field is greater than zero, this method creates an `ccrz__E_OrderTerm__c` that references a `ccrz__E_Term__c` record where `ccrz__Type__c` is Billing.
3. Performs the DML `insert` operation on the `ccrz__E_OrderTerm__c` records.


## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.cc_hk_Invoice.PARAM_ORDER_TERMS`

`List<ccrz__E_OrderTerm__c>` of the inserted order terms.



 **Note:** This output is returned only when the input data specifies `ccrz.ccLogicCartPlace.INVOICE_ENABLED => true`.

SEE ALSO:

[ccrz\\_\\_E\\_OrderTerm\\_\\_c](#)

[ccrz\\_\\_E\\_Term\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartPlace.closeCart

Updates fields on the source `ccrz__E_Cart__c` record to indicate that the cart is closed and a corresponding order was created.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> closeCart (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.processOrderTerms` method completes.

This method requires the following key in the working map:

**ccrz.ccLogicCartPlace.CART\_HEADER**

`ccrz__E_Cart__c` record that represents the source cart.

### Logic Performed

This method completes the following actions:

1. Sets the following fields on the source `ccrz__E_Cart__c` record:

Field	Value
<code>ccrz__ActiveCart__c</code>	<code>false</code>
<code>ccrz__CartStatus__c</code>	<code>Closed</code>

2. Performs the DML `update` operation on the `ccrz__E_Cart__c` record.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)  
[Inserting and Updating Records](#)

## ccrz.ccLogicCartPlace.createStoredPayments

Evaluates whether to create a stored payment from the input payment data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> createStoredPayments (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.closeCart` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiCart.PAYMENTDATA**

JSON-formatted string that describes payment information passed to the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA` input.

### **ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

### **ccrz.ccLogicCartPlace.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create.

## Logic Performed

This method completes the following actions:

1. Checks the value of the My Wallet storefront configuration module's Enabled setting (`wlt.enabled`). If the configuration is `true`, execution continues.
2. Parses `ccrz.ccApiCart.PAYMENTDATA` to determine whether the buyer chose to save the payment information as a stored payment. If so, execution continues.

3. Calls the `ccrz.cc_hk_Payment.createStoredPayment` extension point method for creating a `ccrz__E_StoredPayment__c` record from the payment data.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.createStoredPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

SEE ALSO:

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)  
[ccrz.cc\\_hk\\_Payment.createStoredPayment](#)  
[ccrz.cc\\_hk\\_Payment.processPayment](#)  
[Salesforce Help Configure Payment Options Available on the Checkout Page](#)

## ccrz.ccLogicCartPlace.processSubscriptions

Creates `ccrz__E_Subscription__c` records for any `ccrz__E_SubProdTerm__c` records associated with the order.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processSubscriptions(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.createStoredPayments` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.createStoredPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

### **ccrz.ccLogicCartPlace.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create.

### **'hasSubProdTerm'**

Boolean

Value	Usage
<i>true</i>	The source <code>ccrz__E_Cart__c</code> included a <code>ccrz__E_CartItem__c</code> with an associated <code>ccrz__E_SubProdTerm__c</code> record.
<i>false</i>	The source <code>ccrz__E_Cart__c</code> didn't have subscription data. This method exits without evaluating subscriptions.

## Logic Performed

This method completes the following actions:

1. Evaluates the 'hasSubProdTerm' input.
2. Calls the `ccrz.cc_hk_Subscriptions.onOrder` extension point method for creating the order's `ccrz__E_Subscription__c` records.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccLogicCartPlace.SUB\_RESULT**

`Map<String, Object>` that the `ccrz.cc_hk_Subscriptions.onOrder` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_SubProdTerm\\_\\_c](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz.ccLogicCartPlace.processTransactionPayments

Evaluates whether to call the `ccrz.cc_hk_Order.createTransaction` extension point method, and then calls `ccrz.ccPaymentProcessor` methods for processing the payment.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processTransactionPayments(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.processSubscriptions` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiCart.PAYMENTDATA**

JSON-formatted string that describes payment information passed to the [ccrz.cc\\_hk\\_Payment.processPayment](#) extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA` input.

**ccrz.ccApiCart.PAYMENTRESULT**

Map<String, Object> that represents the output from the [ccrz.cc\\_hk\\_Payment.processPayment](#) extension point method.

**ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the [ccrz.cc\\_hk\\_Payment.createStoredPayment](#) extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

**ccrz.ccLogicCartPlace.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create.

**ccrz.ccLogicCartPlace.SUB\_RESULT**

Map<String, Object> that the `ccrz.cc_hk_Subscriptions.onOrder` method returned.

## Logic Performed

This method completes the following actions:

1. Verifies that `ccrz.ccApiCart.PAYMENTDATA` and `ccrz.ccApiCart.TRANSPAYMENTDATA` exist for the order.
2. Calls the [ccrz.cc\\_hk\\_Order.createTransaction](#) extension point method for creating the order's transaction payment details.
3. Calls the [ccrz.ccPaymentProcessor.process](#) utility method for processing the transaction payment.
4. Calls the [ccrz.ccPaymentProcessor.postProcess](#) utility method for handling the payment's success or failure.

## Outputs

This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccLogicCartPlace.TRANSACTION\_RESULT**

Map<String, Object> that the `ccrz.cc_hk_Order.createTransaction` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)  
[ccrz.cc\\_hk\\_Order.createTransaction](#)  
[ccrz.ccPaymentProcessor.process](#)  
[ccrz.ccPaymentProcessor.postProcess](#)

## ccrz.ccLogicCartPlace.changeOrderStatus

Updates fields on the created `ccrz__E_Order__c` record to indicate that the order was placed.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> changeOrderStatus (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.processTransactionPayments` method completes.

This method requires the following keys in the working map:

### `ccrz.ccLogicCartPlace.CART_HEADER`

`ccrz__E_Cart__c` record that represents the source cart.

### `ccrz.ccLogicCartPlace.ORDER_HEADER`

`ccrz__E_Order__c` record that represents the order to place.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiOrder.PARAM_ORDER_STATUS`

String that specifies a status to set on the order when it's created.

## Logic Performed

This method completes the following actions:

1. Sets the `ccrz__OrderStatus__c` field on the `ccrz__E_Order__c` record, in the following order of precedence:
  - a. The value of the `ccrz.ccApiOrder.PARAM_ORDER_STATUS` input, if specified.
  - b. `On Hold` if the cart or order indicates a tax exemption.
  - c. `Order Submitted` for all other cases.
2. Performs the DML `update` operation on the `ccrz__E_Order__c` record.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[Inserting and Updating Records](#)

## `ccrz.ccLogicCartPlace.processInvoices`

Calls the `ccrz.cc_hk_Invoice.onOrder` extension point method for creating invoice for the order.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processInvoices(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.changeOrderStatus` method completes.

This method requires the following keys in the working map:

### **ccrz.ccLogicCartPlace.CART\_HEADER**

`ccrz__E_Cart__c` record that represents the source cart.

### **ccrz.ccLogicCartPlace.INVOICE\_ENABLED**

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>true</code> .
<code>false</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>false</code> or null.

### **ccrz.ccLogicCartPlace.MAJOR\_ORDER\_ITEMS**

`Map<String, ccrz__E_OrderItem__c>`, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is the `ccrz__E_OrderItem__c` to create for the corresponding major line item in the order.

### **ccrz.ccLogicCartPlace.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to place.

### **ccrz.cc\_hk\_Invoice.PARAM\_CART\_ITEMS**

`List<ccrz__E_CartItem__c>` of the original cart items included in the `ccrz.ccLogicCartPlace.CART_ITEM_FETCH_RESPONSE` input.

### **ccrz.cc\_hk\_Invoice.PARAM\_ORDER\_CHILD\_ITEMS**

`List<ccrz__E_OrderItem__c>` of the order's minor line items.

### **ccrz.cc\_hk\_Invoice.PARAM\_ORDER\_TERMS**

`List<ccrz__E_OrderTerm__c>` of the inserted order terms.

## Logic Performed

This method completes the following actions:

1. Verifies that `ccrz.ccLogicCartPlace.INVOICE_ENABLED` is `true`. Otherwise, the method exits.
2. Passes the other inputs to the `ccrz.cc_hk_Invoice.onOrder` extension point method for creating the `ccrz__E_Invoice__c` record for the order.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Invoice\\_\\_c](#)

[Salesforce Help Generate an Invoice for a Buyer's Order](#)

## ccrz.ccLogicCartPlace.refetch

Call the `ccrz.ccApiCart.fetch` method for fetching the ordered cart.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlace.processInvoices` method.

This method requires the following keys in the working map:

### **ccrz.ccApi.SIZING**

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

### **ccrz.ccApiCart.CART\_ID**

String that specifies the ID of the cart to query.



## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiCart.fetch` outputs.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)


## ccrz.ccLogicCartPlace.placeOrder

Passes the input data to the `ccrz.cc_api_Cart.placeEOrder` public method, which invokes methods of the `ccrz.cc_hk_Order` and `ccrz.cc_hk_Payment` extension point classes. Those methods return data to this method, including the created order and the transaction details.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Important:** This method isn't used in the logic chain for B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12). The `ccrz.cc_api_Cart.placeEOrder` public method is deprecated in this version.

## Signature

```
global virtual Map<String, Object> placeOrder (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlace.processInput` method.

This method requires one of the following keys to execute:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to convert.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to convert.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApi.API\_SIZING**

Map<String, Object> that describes options for [sizing and scoping the method's return data](#). This method supports `ccrz.ccApi.SZ_REFETCH => TRUE`, which specifies that this method invokes `ccrz.ccApiCart.fetch` for returning the ordered cart.

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_REFETCH => TRUE
    }
}
```

**ccrz.ccApiCart.PAYMENTDATA**

JSON-formatted string that describes payment information passed to the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA` input.

**ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

**ccrz.ccApiCart.PAYMENTRESULT**

Map<String, Object> that represents the output from the `ccrz.cc_hk_Payment.processPayment` extension point method.

**ccrz.ccApiOrder.PARAM\_ORDER\_STATUS**


String that specifies the status of the corresponding order. By default, this method expects the value *Order Submitted*.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, which contains one Map<String, Object> that represents the `ccrz__E_Cart__c` record, which the `ccrz.ccApiCart.fetch` method returned.

**Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

**ccrz.ccApiCart.ORDER\_ID**

String that specifies the Salesforce ID of the created `ccrz__E_Order__c`.

**ccrz.ccApiCart.SUBCRESLT**

Map<String, Object> that contains the results of the `ccrz.cc_hk_Subscriptions.onOrder` method, which is invoked from `ccrz.cc_api_Cart.placeEOrder`.

**ccrz.ccApiCart.TRANSACTIONRESULT**

Map<String, Object> that contains the results of the `ccrz.cc_hk_Order.createTransaction` method, which is invoked from `ccrz.cc_api_Cart.placeEOrder`.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[ccrz.cc\\_hk\\_Payment.processPayment](#)

[ccrz.cc\\_hk\\_Order.createTransaction](#)

## ccrz.ccLogicCartPlace.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working Map<String, Object> after the appropriate method completes, depending on which version of the managed package you're using:

#### B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12)

[ccrz.ccLogicCartPlace.refetch](#)

#### B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11) or earlier

[ccrz.ccLogicCartPlace.placeOrder](#)

## Outputs

This method removes the following intermediate keys from the working map, depending on which version of the managed package you're using, and returns the final `Map<String, Object>`:

### B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12)

- `ccrz.ccApiCart.PAYMENTDATA`
- `ccrz.ccApiCart.PAYMENTRESULT`
- `ccrz.ccApiCart.TRANSPAYMENTDATA`
- `ccrz.ccApiOrder.PARAM_ORDER_STATUS`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicCartPlace.CART_HEADER`
- `ccrz.ccLogicCartPlace.INVOICE_ENABLED`
- `ccrz.ccLogicCartPlace.MAJOR_ORDER_ITEMS`
- `ccrz.ccLogicCartPlace.MINOR_ORDER_ITEMS`
- `ccrz.ccLogicCartPlace.ORDER_BILLING_ADDRESS`
- `ccrz.ccLogicCartPlace.ORDER_HEADER`
- `ccrz.ccLogicCartPlace.ORDER_SHIPPING_ADDRESS`
- `ccrz.cc_hk_Invoice.PARAM_CART_ITEMS`
- `ccrz.cc_hk_Invoice.PARAM_ORDER_CHILD_ITEMS`
- `ccrz.cc_hk_Invoice.PARAM_ORDER_TERMS`

### B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11) or earlier

- `ccrz.ccLogic.ORIGINAL_INPUT`

## `ccrz.ccLogicCartPlace.handleCartPlaceException`

Catches a `ccrz.ccLogicCartPlace.CartPlaceException` that occurs during the logic chain, and logs the exception message.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> handleCartPlaceException(Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that can include the following keys:

**'CartPlaceException'**

`ccrz.ccLogicCartPlace.CartPlaceException` that occurred during the `ccrz.ccLogicCartPlace.process` method or its inner methods. The `ccrz.ccLogicCartPlace.process` method rethrows the exception after it's caught, which ends processing and rolls back transactions.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

## ccrz.ccLogicCartPlaceAsync

Define the business logic for converting a cart to an order using asynchronous Apex jobs.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Global API Caller

[ccrz.ccApiCart.placeAsync](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [prepareAsyncOrder](#)

Wraps inner methods that execute synchronously during the initial `ccrz.ccApiCart.placeAsync` call. These methods determine whether the cart satisfies the requirements for placing an order.

#### [retrieveCartHeader](#)

Calls the `ccrz.ccApiCart.fetch` method for querying the `ccrz__E_Cart__c` record to place an order from. This method can also evaluate the cart's `ccrz__ValidationStatus__c` before creating an order. Finally, this method also queries the cart's related `ccrz__E_CartCoupon__c` and `ccrz__E_ContactAddr__c` records, but doesn't query the cart's items. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

#### [processAddresses](#)

Copies the cart's billing and shipping addresses, so that the copies can apply to the order that this logic chain creates. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

### [createOrderHeader](#)

Creates the `ccrz__E_Order__c` record by copying all relevant field values from the source `ccrz__E_Cart__c` record. This method doesn't perform the DML `insert` operation. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

### [performDMLOnOrder](#)

Performs the DML `insert` operation on the order's billing and shipping addresses. Then, this method updates field values on the `ccrz__E_Cart__c` and `ccrz__E_Order__c` records, and performs the DML `update` operation on those records. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

### [validateCartPricing](#)

Evaluates whether the cart's price data is current. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

### [enqueuePrice](#)

Calls the `System.enqueueJob` method for adding the queueable `ccLogicCartPlaceEnqueuePricing` class to the asynchronous Apex job queue. To call a custom queueable class, override this method with custom subscriber code. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

### [enqueuePlace](#)

Calls the `System.enqueueJob` method for adding the queueable `ccLogicCartPlaceEnqueue` class to the asynchronous Apex job queue. To call a custom queueable class, override this method with custom subscriber code.

### [executeAsyncOrder](#)

Wraps inner methods that execute during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` calls. These methods process a subset of `ccrz__E_CartItem__c` records from the source `ccrz__E_Cart__c` and converts them to `ccrz__E_OrderItem__c` records.

### [checkForInvalidItems](#)

Evaluates whether the source `ccrz__E_Cart__c` contains `ccrz__E_CartItem__c` records where the `ccrz__ItemStatus__c` field's value is `Invalid`. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### [retrieveCartItems](#)

Determines how many `ccrz__E_CartItem__c` records to process during the recursive job, and then calls the `ccrz.ccApiCart.fetchCartItems` method for querying them. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### [processCartItems](#)

Separates the batch of fetched `ccrz__E_CartItem__c` records into major and minor line items, and calls the `ccrz.ccLogicCartPlaceAsync.createOrderItem` method for creating corresponding `ccrz__E_OrderItem__c` records. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### [createOrderItem](#)

Creates a `ccrz__E_OrderItem__c` record by copying all relevant field values from a source `ccrz__E_CartItem__c` record. This method also evaluates the source cart item's groups and subscriptions, and sorts the order item by whether it's a major or minor line item. The `ccrz.ccLogicCartPlaceAsync.processCartItems` method calls this method for each `ccrz__E_CartItem__c` associated with the source cart.

### [processOrderGroups](#)

Clones shipping addresses from source `ccrz__E_CartItemGroup__c` records for the corresponding `ccrz__E_OrderItemGroup__c` records. This method also inserts the created addresses and order item groups, and finally associates the order item groups with their corresponding order items. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### [performDMLOnOrderItems](#)

Calls inner methods for performing the DML `insert` operation for the order's major and minor line items, and then evaluates whether the cart has a coupon applied. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### [insertMajorItems](#)

Sets the `ccrz__Order__c` and `ccrz__RequestedDate__c` fields on each `ccrz__E_OrderItem__c` record that represents a major line item in the order. Then, the method performs the DML `insert` operation on the `ccrz__E_OrderItem__c` records. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### [insertMinorOrderItems](#)

Sets the `ccrz__Order__c`, `ccrz__RequestedDate__c`, and `ccrz__ParentOrderItem__c` fields on each `ccrz__E_OrderItem__c` record that represents a minor line item in the order. Then, the method performs the DML `insert` operation on the `ccrz__E_OrderItem__c` records. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### [enqueueNext](#)

Determines whether more major line item `ccrz__E_CartItem__c` records remain to be processed, and calls `ccrz.ccLogicCartPlaceAsync.enqueuePlace` for queueing the next job. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### [processOrderTerms](#)

Creates and inserts `ccrz__E_OrderTerm__c` records for the order. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

### [closeCart](#)

Updates fields on the source `ccrz__E_Cart__c` record to indicate that the cart is closed and a corresponding order was created. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

### [createStoredPayments](#)

Evaluates whether to create a stored payment from the input payment data. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

### [processSubscriptions](#)

Creates `ccrz__E_Subscription__c` records for any `ccrz__E_SubProdTerm__c` records associated with the order. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

### [processTransactionPayments](#)

Evaluates whether to call the `ccrz.cc_hk_Order.createTransaction` extension point method, and then calls `ccrz.ccPaymentProcessor` methods for processing the payment. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

### [changeOrderStatus](#)

Updates fields on the created `ccrz__E_Order__c` record to indicate that the order was placed. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

### [processInvoices](#)

Calls the `ccrz.cc_hk_Invoice.onOrder` extension point method for creating invoice for the order. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

### [sendOrderConfirmation](#)

Calls the `System.enqueueJob` method for adding the queueable `ccrz.cc_util_OrderConfirmationEmail.ccUtilOrderConfirmationEmail` class to the asynchronous Apex job queue for sending an order confirmation email. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

### [updateOrderOnError](#)

Updates the status of the `ccrz__E_Order__c` record and deletes the order's billing or shipping `ccrz__E_ContactAddr__c` records when an exception occurs during asynchronous order placement. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### [sendOrderFail](#)

Calls the `System.enqueueJob` method for adding the queueable `ccrz.cc_util_OrderConfirmationEmail.ccUtilOrderConfirmationEmail` class to the asynchronous Apex job queue for sending an order failure email.

### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys. This method executes for every `ccrz.ccApiCart.placeAsync` call, regardless of whether it's the initial or a recursive call.

### [handleCartPlaceException](#)

Catches a `ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException` that occurs during the logic chain, and logs the exception message.

## ccrz.ccLogicCartPlaceAsync.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.placeAsync` method.

## Logic Performed

Asynchronous order placement consists of calls to the `ccrz.ccApiCart.placeAsync` method that occur in two phases:




### Initial synchronous call

The initial `ccrz.ccApiCart.placeAsync` call enters a *synchronous* portion, where this method calls into the following inner methods:

1. `processInput`
2. `prepareAsyncOrder`, which calls other inner methods

### Recursive, asynchronous calls

Each recursive `ccrz.ccApiCart.placeAsync` call enters an *asynchronous* portion, where this method calls into `executeAsyncOrder`, which calls other inner methods.

 **Note:** The working map maintains an input key, `ccrz.ccApiCart.PLACE_ASYNC_FINISH`, that indicates the status of the logic chain. When this key is `false`, the logic is executing the synchronous portion. When this key becomes `true`, the logic is executing the asynchronous portion.

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

This method rethrows any `ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException` instances that occur during the logic chain, which ends processing and rolls back transactions. The

`ccrz.ccLogicCartPlaceAsync.handleCartPlaceException` method catches these exceptions.

## ccrz.ccLogicCartPlaceAsync.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlaceAsync.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the cart to convert.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to convert.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApi.SUCCESS**

Boolean that's assigned an intermediate value of `false`. Subsequent methods in the logic chain update this value to `true` when the logic completes successfully.

### **ccrz.ccApiCart.ENFORCE\_PRICE\_VALIDATION\_ON\_PLACE**

Boolean that matches the value of the `ccrz.ccApiCart.ENFORCE_PRICE_VALIDATION_ON_PLACE` input. If this key wasn't specified in the input data, this method sets a fallback value:

Value	Usage
<code>true</code>	The Cart Scale storefront configuration module's Enable Cart Scale setting ( <code>cs.enable</code> ) is <code>true</code> .
<code>false</code>	The Cart Scale storefront configuration module's Enable Cart Scale setting ( <code>cs.enable</code> ) is <code>false</code> or <code>null</code> .

### **ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

### **ccrz.ccLogicCartPlaceAsync.INVOICE\_ENABLED**

Boolean

Value	Usage
<code>true</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>true</code> .
<code>false</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>false</code> or <code>null</code> .

The `ccrz.ccLogicCartPlace.prepReturn` method removes these intermediate keys from the working map.

SEE ALSO:

[ccrz.ccApiCart.placeAsync](#)

## ccrz.ccLogicCartPlaceAsync.prepareAsyncOrder

Wraps inner methods that execute synchronously during the initial `ccrz.ccApiCart.placeAsync` call. These methods determine whether the cart satisfies the requirements for placing an order.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> prepareAsyncOrder (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.processInput` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiCart.PLACE_ASYNC_FINISH`

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	The current method execution is a recursive, asynchronous call that continues the order placement process after the initial call.
<code>false</code>	The current method execution is the synchronous portion of the initial call that begins the order placement process.

## Logic Performed

During this synchronous call, this method invokes other methods of the `ccrz.ccLogicCartPlaceAsync` class in the following order:

1. `retrieveCartHeader`
2. `processAddresses`
3. `createOrderHeader`
4. `performDMLOnOrder`
5. `validateCartPricing`
6. `enqueuePlace` when the cart's price data is current or `enqueuePrice` when the cart's price data isn't current
7. `prepReturn`

## Outputs

This method returns the working `Map<String, Object>` after all inner methods have completed. Refer to the individual method reference for more information about each method's outputs.

SEE ALSO:

[ccrz.ccApiCart.placeAsync](#)

## ccrz.ccLogicCartPlaceAsync.retrieveCartHeader

Calls the `ccrz.ccApiCart.fetch` method for querying the `ccrz__E_Cart__c` record to place an order from. This method can also evaluate the cart's `ccrz__ValidationStatus__c` before creating an order. Finally, this method also queries the cart's related `ccrz__E_CartCoupon__c` and `ccrz__E_ContactAddr__c` records, but doesn't query the cart's items. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> retrieveCartHeader(Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlaceAsync.prepareAsyncOrder` method.


This method requires either of the following keys in the working map:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to convert.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to convert.

 **Note:** If neither key is specified, this method returns a `ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException`.

### Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiCart.SKIP\_VALID\_CHECK**

Boolean

Value	Usage
<i>true</i>	During this method's execution, ignore the cart's <code>ccrz__ValidationStatus__c</code> field and attempt to create an order from the cart.
<i>false</i> (default)	During this method's execution, verify that the cart's <code>ccrz__ValidationStatus__c</code> field is set before creating an order for this cart. If the cart isn't valid, this method returns a <code>ccrz.cc_api_Cart.CartValidationException</code> .

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiCart.fetch` outputs, and sets the following intermediate keys:

### `ccrz.ccApiAddress.ADDRESSLIST`

`List<ccrz__E_ContactAddr__c>` of the cart's shipping or billing addresses. For more information about fetching address data with cart data, see [ccrz.ccApiCart.fetch](#).

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the fetched cart.

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the fetched cart.

### `ccrz.ccApiCart.HAS_COUPON_CODE`

Boolean

Value	Usage
true	The cart has a coupon applied.
false	The cart doesn't have any coupon applied.

### `ccrz.ccLogicCartPlaceAsync.CART_HEADER`

`ccrz__E_Cart__c` record that represents the cart to convert to an order.

### `ccrz.ccLogicCartPlaceAysnc.COUPON_TO_UPDATE`

`List<ccrz__E_CartCoupon__c>` that contains a coupon associated with the cart.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## `ccrz.ccLogicCartPlaceAsync.processAddresses`

Copies the cart's billing and shipping addresses, so that the copies can apply to the order that this logic chain creates. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processAddresses (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.retrieveCartHeader` method completes.

This method requires the following keys in the working map:

### `ccrz.ccApiAddress.ADDRESSLIST`

`List<ccrz__E_ContactAddr__c>` of the cart's shipping or billing addresses.

### `ccrz.ccLogicCartPlaceAsync.CART_HEADER`

`ccrz__E_Cart__c` record that represents the cart to convert to an order.

## Logic Performed

This method completes the following actions:

1. Parses `ccrz.ccApiAddress.ADDRESSLIST` for the cart's `ccrz__BillTo__c` billing address, and clones the corresponding `ccrz__E_ContactAddr__c` record.
2. Parses `ccrz.ccApiAddress.ADDRESSLIST` for the cart's `ccrz__ShipTo__c` shipping address, and clones the corresponding `ccrz__E_ContactAddr__c` record.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccLogicCartPlaceAsync.ORDER_BILLING_ADDRESS`

`ccrz__E_ContactAddr__c` record cloned from the cart's billing address.

### `ccrz.ccLogicCartPlaceAsync.ORDER_SHIPPING_ADDRESS`

`ccrz__E_ContactAddr__c` record cloned from the cart's shipping address.

This method also removes the following intermediate key from the working map:

- `ccrz.ccApiAddress.ADDRESSLIST`

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

## `ccrz.ccLogicCartPlaceAsync.createOrderHeader`

Creates the `ccrz__E_Order__c` record by copying all relevant field values from the source `ccrz__E_Cart__c` record. This method doesn't perform the DML `insert` operation. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> createOrderHeader (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.processAddresses` method completes.

This method requires the following key in the working map:

**ccrz.ccLogicCartPlaceAsync.CART\_HEADER**  
`ccrz__E_Cart__c` record that represents the cart to convert to an order.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**  
`ccrz__E_Order__c` record with relevant fields copied from the source `ccrz__E_Cart__c` record.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)  
[ccrz\\_\\_E\\_Order\\_\\_c](#)

## ccrz.ccLogicCartPlaceAsync.performDMLOnOrder

Performs the DML `insert` operation on the order's billing and shipping addresses. Then, this method updates field values on the `ccrz__E_Cart__c` and `ccrz__E_Order__c` records, and performs the DML `update` operation on those records. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> performDMLOnOrder (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.createOrderHeader` method completes.

This method requires the following keys in the working map:

**ccrz.ccLogicCartPlaceAsync.CART\_HEADER**

`ccrz__E_Cart__c` record that represents the cart to convert to an order.

**ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**


`ccrz__E_Order__c` record that represents the order to insert.

**'orderBillingAddress'**

`ccrz__E_ContactAddr__c` record cloned from the order's billing address to insert.

**'orderShippingAddress'**

`ccrz__E_ContactAddr__c` record cloned from the order's shipping address to insert.

 **Note:** The `ccrz.ccLogicCartPlaceAsync.processAddresses` method cloned the cart's billing and shipping addresses to create these `ccrz__E_ContactAddr__c` records.

## Logic Performed

This method completes the following actions:

1. Performs the DML `insert` operation on the `ccrz__E_ContactAddr__c` records for the order's billing and shipping addresses
2. Associates the inserted addresses with the `ccrz__E_Order__c` record.
3. Sets the `ccrz__OrderStatus__c` field on the `ccrz__E_Order__c` record to `In Creation`.
4. Performs the DML `insert` operation on the `ccrz__E_Order__c` record.
5. Sets the following fields on the `ccrz__E_Cart__c` record:

Field	Value
<code>ccrz__ActiveCart__c</code>	<code>false</code>
<code>ccrz__CartStatus__c</code>	<code>Processing</code>

6. Performs the DML `update` operation on the `ccrz__E_Cart__c` record.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiCart.ORDER\_ID**

String that specifies the Salesforce ID of the inserted `ccrz__E_Order__c` record.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz.ccLogicCartPlaceAsync.processAddresses](#)

[Inserting and Updating Records](#)



## ccrz.ccLogicCartPlaceAsync.validateCartPricing

Evaluates whether the cart's price data is current. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> validateCartPricing(Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.performDMLOnOrder` method completes.


This method requires the following key in the working map:

**ccrz.ccLogicCartPlaceAsync.CART\_HEADER**  
`ccrz__E_Cart__c` record that represents the cart to convert to an order.


### Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccApiCart.ALLOW\_PRICING\_ON\_ASYNC\_ORDER**  
 Boolean

Value	Usage
<code>true</code>	<p>If the <code>ccrz.ccApiCart.ENFORCE_PRICE_VALIDATION_ON_PLACE</code> input is <code>true</code> and the cart's price data isn't current, call the <code>ccrz.ccApiCart.priceAsync</code> method to reprice the cart.</p> <p> <b>Note:</b> This behavior can cause issues with the cart's shipping and taxes, which are calculated based on the cart's previous price data, not the repriced cart. The <code>ccrz.ccApiCart.priceAsync</code> method doesn't update shipping or taxes. Consider creating an extension that updates shipping and tax calculations for asynchronous pricing.</p>
<code>false</code> (default)	If the cart's price data isn't current, the cart isn't repriced, and the order placement process fails.

**ccrz.ccApiCart.ENFORCE\_PRICE\_VALIDATION\_ON\_PLACE**  
 Boolean

Value	Usage
<code>true</code>	Validate that the cart's price data is current before starting the order placement process.  <b>Note:</b> When you enable Cart Scale, this input is <code>true</code> by default.
<code>false</code> (default)	Don't validate that the cart's price data is current before starting order placement.

## Logic Performed

This method completes the following actions:


- Checks whether the input data includes the `ccrz.ccApiCart.ENFORCE_PRICE_VALIDATION_ON_PLACE` key with a value of `true`:
  - If so, this method continues.
  - If the input's value is `false` or not specified, this method exits.
- Evaluates the cart's `ccrz__RepricedDate__c` field.
  - If the cart's price data is current, this method then checks whether the input data includes the `ccrz.ccApiCart.ALLOW_PRICING_ON_ASYNC_ORDER` key with a value of `true`. If so, this method sets the `ccrz.ccLogicCartPlaceAsync.PRICE_IS_STALE` output to `true`.
  - If the cart's price data isn't current, this method returns a `ccrz.cc_api_Cart.CartValidationException`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**`ccrz.ccLogicCartPlaceAsync.PRICE_IS_STALE`**

Boolean

Value	Usage
<code>true</code>	The cart's price data isn't current.  <b>Note:</b> By default, this output is <code>true</code> only when calling the <code>ccrz.ccApiCart.priceAsync</code> method is required.
<code>false</code>	The cart's price data is current.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz.ccApiCart.priceAsync](#)

## `ccrz.ccLogicCartPlaceAsync.enqueuePrice`

Calls the `System.enqueueJob` method for adding the queueable `ccLogicCartPlaceEnqueuePricing` class to the asynchronous Apex job queue. To call a custom queueable class, override this method with custom subscriber code. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.placeAsync` method call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> enqueuePrice (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.validateCartPricing` method completes.

This method requires the following key in the working map:

### `ccrz.ccLogicCartPlaceAsync.PRICE_IS_STALE`

Boolean that must be `true` for this method to execute.

Value	Usage
<code>true</code>	The cart's price data isn't current.
<code>false</code>	The cart's price data is current.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCart.PLACE_ASYNC_JOB_ID`

ID of the `AsyncApexJob` that represents the asynchronous Apex job that the `System.enqueueJob` method returned.

SEE ALSO:

[enqueueJob \(queueableObj\)](#)

[Queueable Apex](#)

## `ccrz.ccLogicCartPlaceAsync.enqueuePlace`

Calls the `System.enqueueJob` method for adding the queueable `ccLogicCartPlaceEnqueue` class to the asynchronous Apex job queue. To call a custom queueable class, override this method with custom subscriber code.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> enqueuePlace (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.validateCartPricing` method completes.

This method requires the following key in the working map:

### **ccrz.ccLogicCartPlaceAsync.PRICE\_IS\_STALE**

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	The cart's price data isn't current.
<code>false</code>	The cart's price data is current.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiCart.PLACE\_ASYNC\_JOB\_ID**

ID of the `AsyncApexJob` that represents the asynchronous Apex job that the `System.enqueueJob` method returned.

SEE ALSO:

[enqueueJob \(queueableObj\)](#)

[Queueable Apex](#)

## ccrz.ccLogicCartPlaceAsync.executeAsyncOrder

Wraps inner methods that execute during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` calls. These methods process a subset of `ccrz__E_CartItem__c` records from the source `ccrz__E_Cart__c` and converts them to `ccrz__E_OrderItem__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> executeAsyncOrder (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.processInput` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiCart.PLACE_ASYNC_FINISH`

Boolean that must be `true` for this method to execute.


Value	Usage
<code>true</code>	The current method execution is a recursive, asynchronous call that continues the order placement process after the initial call.
<code>false</code>	The current method execution is the synchronous portion of the initial call that begins the order placement process.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiCart.PRICING_FOR_ORDER_FAILED`


Boolean

Value	Usage
<code>true</code>	<p>An error occurred when pricing the cart during the previous iteration of the recursive, asynchronous call. The logic chain returns a <code>ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException</code>, and the <code>ccrz.ccLogicCartPlaceAsync.updateOrderOnError</code> method executes.</p> <p> <b>Note:</b> This input is used only when the <code>ccrz.ccApiCart.ALLOW_PRICING_ON_ASYNC_ORDER</code> input is <code>true</code> and the cart was repriced because its price data wasn't current.</p>
<code>false</code> (default)	No error occurred.


## Logic Performed

During each recursive, asynchronous call, this method invokes other methods of the `ccrz.ccLogicCartPlaceAsync` class in the following order for converting a subset of `ccrz__E_CartItem__c` records to corresponding `ccrz__E_OrderItem__c` records:

1. `checkForInvalidItems`
2. `retrieveCartItems`

 **Note:** By default, B2B Commerce for Visualforce defines a batch size of 200 `ccrz__E_CartItem__c` records that represent major line items and coupons. The batch size doesn't consider how many minor line items are associated with the major line items.


3. `processCartItems`, which calls into `createOrderItem`

 **Note:** The logic chain specifies a database savepoint after the `processCartItem` method completes to allow for rollback when errors occur. This rollback applies only to the batch of items that the current job is processing, and not *all* items.

4. `processOrderGroups`
5. `performDMLOnOrderItems`, which calls into `insertMajorItems` and `insertMinorOrderItems`
6. `enqueueNext`, which calls into `enqueuePlace`
7. `prepReturn`

During the final recursive `ccrz.ccApiCart.placeAsync` call that processes the remaining `ccrz__E_CartItem__c` records, this method invokes the remaining methods of the `ccrz.ccLogicCartPlaceAsync` class in the following order:

1. `processOrderTerms`
2. `closeCart`
3. `createStoredPayments`
4. `processSubscriptions`
5. `processTransactionPayments`
6. `changeOrderStatus`
7. `processInvoices`
8. `sendOrderConfirmation`
9. `prepReturn`

 **Note:** If any exceptions occur during asynchronous processing, the `updateOrderOnError` and `sendOrderFail` methods execute.

## Outputs

This method returns the working `Map<String, Object>` after all inner methods have completed. Refer to the individual method reference for more information about each method's outputs.

## `ccrz.ccLogicCartPlaceAsync.checkForInvalidItems`

Evaluates whether the source `ccrz__E_Cart__c` contains `ccrz__E_CartItem__c` records where the `ccrz__ItemStatus__c` field's value is `Invalid`. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> checkForInvalidItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlaceAsync.executeAsyncOrder` method.

This method requires the following key in the working map:

### **ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to convert.

## Logic Performed

If the source `ccrz__E_Cart__c` contains `ccrz__E_CartItem__c` records where the `ccrz__ItemStatus__c` field's value is `Invalid`, this method returns a `ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException`.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartPlaceAsync.retrieveCartItems

Determines how many `ccrz__E_CartItem__c` records to process during the recursive job, and then calls the `ccrz.ccApiCart.fetchCartItems` method for querying them. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> retrieveCartItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.checkForInvalidItems` method completes.

This method requires the following key in the working map:

### **ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to query items for.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### **ccrz.ccApiCart.CARTITEMLASTID**

String that specifies the Salesforce ID of the final major cart item that was processed during the previous recursive job.

### **ccrz.ccApiCart.CART\_PLACE\_BATCH\_SIZE**

Integer that specifies the number of major cart items to process during each recursive job.



#### Note:

- If you don't specify a value for this input, the method defaults to 200 major cart items.
- The job also queries any minor cart items that are associated with the major cart items. As a result, the total number of processed `ccrz__E_CartItem__c` records is higher than the specified batch size when the cart contains complex products.

## Logic Performed

This method completes the following actions:

1. Evaluates whether the input data already specifies a value for `ccrz.ccApiCart.CART_PLACE_BATCH_SIZE`. Otherwise, the method sets a default value of 200 `ccrz__E_CartItem__c` records to evaluate during the recursive job.
2. Evaluates whether the input data specifies `ccrz.ccApiCart.CARTITEMLASTID`:
  - If so, the method starts processing `ccrz__E_CartItem__c` records from the final ID evaluated during the previous job.
  - Otherwise, the method starts processing `ccrz__E_CartItem__c` records from the beginning of the `ccrz__E_Cart__c` record's list of items.
3. Executes a direct query of major item `ccrz__E_CartItem__c` records for the source `ccrz__E_Cart__c`.



**Note:** To keep the order of `ccrz__E_CartItem__c` records consistent for subsequent recursive jobs, the query uses a `SQL ORDER BY Id` clause.

4. Calls the `ccrz.ccApiCart.fetchCartItems` method for returning the batch of major cart items for the current job.



**Note:** The job also queries any minor cart items that are associated with the major cart items. As a result, the total number of processed `ccrz__E_CartItem__c` records is higher than the specified batch size when the cart contains complex products.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiCart.CARTITEMLASTID**

String that specifies the Salesforce ID of the final major cart item that was processed during the current recursive job.



**ccrz.ccLogicCartPlaceAsync.CART\_ITEM\_FETCH\_RESPONSE**

Map<String, Object> that contains the major and minor [ccrz\\_\\_E\\_CartItem\\_\\_c](#) line items that the `ccrz.ccApiCart.fetchCartItems` returned.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccApiCart.fetchCartItems](#)

**ccrz.ccLogicCartPlaceAsync.processCartItems**

Separates the batch of fetched `ccrz__E_CartItem__c` records into major and minor line items, and calls the `ccrz.ccLogicCartPlaceAsync.createOrderItem` method for creating corresponding `ccrz__E_OrderItem__c` records. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

**Signature**

```
global virtual Map<String, Object> processCartItems (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> after the [ccrz.ccLogicCartPlaceAsync.retrieveCartItems](#) method completes.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the source cart.

**ccrz.ccLogicCartPlaceAsync.CART\_HEADER**

[ccrz\\_\\_E\\_Cart\\_\\_c](#) record that represents the source cart.

**ccrz.ccLogicCartPlaceAsync.CART\_ITEM\_FETCH\_RESPONSE**

Map<String, Object> that contains the [ccrz.ccApiCart.fetchCartItems](#) output, including the [ccrz\\_\\_E\\_CartItem\\_\\_c](#) records associated with the source cart.



**Note:** If this key doesn't reference any cart items, this method returns a `ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException`.

**ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**

[ccrz\\_\\_E\\_Order\\_\\_c](#) record that represents the order to create items for.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### **ccrz.ccLogicCartPlaceAsync.INVOICE\_ENABLED**

Boolean

Value	Usage
<i>true</i>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>true</code> .
<i>false</i>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>false</code> or <code>null</code> .

## Logic Performed

This method completes the following actions:

1. Parses `ccrz.ccLogicCartPlace.CART_ITEM_FETCH_RESPONSE` for `ccrz__E_CartItem__c` records and sorts the items into separate maps of major and minor line items.
2. Checks for any `ccrz__E_CartItemGroup__c` records from the source cart. If cart item groups exist, this method sets up corresponding `ccrz__E_OrderItemGroup__c` records, and builds maps that relate each group's items and groups for subsequent processing:
  - One map, 'cartGroupToOrderItemGroup', that maps *every* source `ccrz__E_CartItemGroup__c`—even groups that are empty and don't reference any individual `ccrz__E_CartItem__c` records—to its corresponding `ccrz__E_OrderItemGroup__c`.
  - A second map, 'orderIGroupsToAdd', that maps only `ccrz__E_CartItemGroup__c` records that aren't empty—that reference individual `ccrz__E_CartItem__c` records—with their corresponding `ccrz__E_OrderItemGroup__c` records. Later, the `ccrz.ccLogicCartPlaceAsync.processOrderGroups` method inserts only the `ccrz__E_OrderItemGroup__c` records that aren't empty.
  - A third map, 'cartGroupToOrderItems', that's initially empty.
  - Maps for managing the cart item group shipping addresses.
3. Calls the `ccrz.ccLogicCartPlaceAsync.createOrderItem` inner method for each `ccrz__E_CartItem__c`, which completes the following actions:
  - Creates a corresponding `ccrz__E_OrderItem__c`.
  - Associates the `ccrz__E_CartItem__c` with the appropriate `ccrz__E_OrderItemGroup__c` in 'orderIGroupsToAdd'.
  - Associates the source `ccrz__E_CartItemGroup__c` with the list of created `ccrz__E_OrderItem__c` records in 'cartGroupToOrderItems'.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccLogicCartPlaceAsync.MAJOR\_ORDER\_ITEMS**

`Map<String, ccrz__E_OrderItem__c>`, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is the `ccrz__E_OrderItem__c` to create for the corresponding major line item in the order.


**ccrz.ccLogicCartPlaceAsync.MINOR\_ORDER\_ITEMS**

Map<String, List<ccrz\_\_E\_OrderItem\_\_c>>, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is a List<ccrz\_\_E\_OrderItem\_\_c> to create as minor line items for the `ccrz__E_OrderItem__c` that corresponds to the major line item from the cart.

**ccrz.cc\_hk\_Invoice.PARAM\_CART\_ITEMS**

List<ccrz\_\_E\_CartItem\_\_c> of the original cart items included in the `ccrz.ccLogicCartPlaceAsync.CART_ITEM_FETCH_RESPONSE` input.

 **Note:** This output is returned only when the input data specifies `ccrz.ccLogicCartPlaceAsync.INVOICE_ENABLED => true`.

**'addressToCartGroupMap'**

Map<ID, ID>, where:

- Each key is the Salesforce ID of the `ccrz__E_ContactAddr__c` record that the `ccrz__E_CartItemGroup__c` record's `ccrz__ShipTo__c` field references.
- Each value is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.

**'cartGroupAddressMap'**

Map<ID, ID>, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the Salesforce ID of the `ccrz__E_ContactAddr__c` record that the `ccrz__E_CartItemGroup__c` record's `ccrz__ShipTo__c` field references.

**'cartGroupToOrderItemGroup'**

Map<ID, `ccrz__E_OrderItemGroup__c`> that maps *every* source `ccrz__E_CartItemGroup__c`—even groups that are empty and don't reference any individual `ccrz__E_CartItem__c` records—to its corresponding `ccrz__E_OrderItemGroup__c`:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the corresponding `ccrz__E_OrderItemGroup__c` record that the method created.

**'cartGroupToOrderItems'**

Map<ID, List<ccrz\_\_E\_OrderItem\_\_c>>, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is a List<ccrz\_\_E\_OrderItem\_\_c> of the order items that the `ccrz.ccLogicCartPlaceAsync.createOrderItem` created and included in the group.

**'orderIGroupsToAdd'**

Map<ID, `ccrz__E_OrderItemGroup__c`> that maps only `ccrz__E_CartItemGroup__c` records that aren't empty—that reference individual `ccrz__E_CartItem__c` records—with their corresponding `ccrz__E_OrderItemGroup__c` records:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.

- Each value is the corresponding `ccrz__E_OrderItemGroup__c` record that the method created.

## SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)[ccrz\\_\\_E\\_OrderItemGroup\\_\\_c](#)[ccrz.ccLogicCartPlaceAsync.createOrderItem](#)

## ccrz.ccLogicCartPlaceAsync.createOrderItem

Creates a `ccrz__E_OrderItem__c` record by copying all relevant field values from a source `ccrz__E_CartItem__c` record. This method also evaluates the source cart item's groups and subscriptions, and sorts the order item by whether it's a major or minor line item. The `ccrz.ccLogicCartPlaceAsync.processCartItems` method calls this method for each `ccrz__E_CartItem__c` associated with the source cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> createOrderItem(Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlaceAsync.processCartItems` method.

This method requires the following keys in the working map:

**'cartItemToCreate'**

`ccrz__E_CartItem__c` record to create an `ccrz__E_OrderItem__c` from.

**'hasCurrencyIsoCode'**

Boolean

Value	Usage
<code>true</code>	The org supports multiple currencies, which indicates that the created <code>ccrz__E_OrderItem__c</code> must specify a value for the <code>currencyIsoCode</code> field.
<code>false</code> (default)	The storefront supports only one currency.

**'majorOrderItemMap'**

Map<String, ccrz\_\_E\_OrderItem\_\_c>, where:

- Each key is the Salesforce ID of a source ccrz\_\_E\_CartItem\_\_c record that represents a major line item in the cart.
- Each value is the ccrz\_\_E\_OrderItem\_\_c to create for the corresponding major line item in the order.

**'minorOrderItemMap'**

Map<String, List<ccrz\_\_E\_OrderItem\_\_c>>, where:

- Each key is the Salesforce ID of a source ccrz\_\_E\_CartItem\_\_c record that represents a major line item in the cart.
- Each value is a List<ccrz\_\_E\_OrderItem\_\_c> to create as minor line items for the ccrz\_\_E\_OrderItem\_\_c that corresponds to the major line item from the cart.

**'workingMap'**

Map<String, Object> that was passed to the ccrz.ccLogicCartPlaceAsync.processCartItems method and then to this method.

## Logic Performed

This method completes the following actions:

1. Creates an ccrz\_\_E\_OrderItem\_\_c from the source ccrz\_\_E\_CartItem\_\_c record.
2. Evaluates whether the source ccrz\_\_E\_CartItem\_\_c has an associated ccrz\_\_E\_SubProdTerm\_\_c.
3. Evaluates the order item's ccrz\_\_OrderLineType\_\_c and sorts the item into an appropriate map:
  - If the order item is a major line item or coupon, the method adds the order item to the map of major items.
  - Otherwise, the method adds the order item to the map of minor items.
4. Evaluates whether the cart item belonged to any ccrz\_\_E\_CartItemGroup\_\_c records, and processes the group information:
  - Associates the ccrz\_\_E\_CartItem\_\_c with the appropriate ccrz\_\_E\_OrderItemGroup\_\_c in the 'orderIGroupsToAdd' key in the working map.
  - Associates the source ccrz\_\_E\_CartItemGroup\_\_c with the list of created ccrz\_\_E\_OrderItem\_\_c records in the 'cartGroupToOrderItems' key in the working map.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**'hasSubProdTerm'**

Boolean

Value	Usage
true	The source ccrz__E_CartItem__c had an associated ccrz__E_SubProdTerm__c record.

Value	Usage
false	The source <code>ccrz__E_CartItem__c</code> didn't have subscription data.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[ccrz.ccLogicCartPlaceAsync.processCartItems](#)

## ccrz.ccLogicCartPlaceAsync.processOrderGroups

Clones shipping addresses from source `ccrz__E_CartItemGroup__c` records for the corresponding `ccrz__E_OrderItemGroup__c` records. This method also inserts the created addresses and order item groups, and finally associates the order item groups with their corresponding order items. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> processOrderGroups (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the [ccrz.ccLogicCartPlaceAsync.processCartItems](#) method completes.

This method requires the following keys in the working map:

#### **ccrz.ccLogicCartPlaceAsync.CART\_HEADER**

`ccrz__E_Cart__c` record that represents the source cart.

#### **ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create.

#### **'addressToCartGroupMap'**

Map<ID, ID>, where:

- Each key is the Salesforce ID of the `ccrz__E_ContactAddr__c` record that the `ccrz__E_CartItemGroup__c` record's `ccrz__ShipTo__c` field references.
- Each value is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.

#### **'cartGroupAddressMap'**

Map<ID, ID>, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the Salesforce ID of the `ccrz__E_ContactAddr__c` record that the `ccrz__E_CartItemGroup__c` record's `ccrz__ShipTo__c` field references.

#### 'cartGroupToOrderItemGroup'

Map<ID, `ccrz__E_OrderItemGroup__c`> that maps *every* source `ccrz__E_CartItemGroup__c`—even groups that are empty and don't reference any individual `ccrz__E_CartItem__c` records—to its corresponding `ccrz__E_OrderItemGroup__c`:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the corresponding `ccrz__E_OrderItemGroup__c` record.

#### 'cartGroupToOrderItems'

Map<ID, List<`ccrz__E_OrderItem__c`>>, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is a List<`ccrz__E_OrderItem__c`> of the order items that the `ccrz.ccLogicCartPlaceAsync.createOrderItem` created and included in the group.

#### 'orderIGroupsToAdd'

Map<ID, `ccrz__E_OrderItemGroup__c`> that maps only `ccrz__E_CartItemGroup__c` records that aren't empty—that reference individual `ccrz__E_CartItem__c` records—with their corresponding `ccrz__E_OrderItemGroup__c` records:

- Each key is the Salesforce ID of a source `ccrz__E_CartItemGroup__c` record.
- Each value is the corresponding `ccrz__E_OrderItemGroup__c` record.

## Logic Performed

This method completes the following actions:

1. Calls the `ccrz.ccApiAddress.fetch` method for querying the `ccrz__E_ContactAddr__c` records of the shipping addresses associated with each `ccrz__E_CartItemGroup__c`.
2. Calls the `ccrz.ccApiAddress.createAddress` method for cloning the fetched shipping addresses.
3. Assigns the appropriate shipping address to each corresponding `ccrz__E_OrderItemGroup__c`.
4. Performs the DML `insert` operation for each `ccrz__E_OrderItemGroup__c` record in 'orderIGroupsToAdd'. These groups reference the individual `ccrz__E_OrderItem__c` records to create.
5. Sets the `ccrz__OrderItemGroup__c` field on each `ccrz__E_OrderItem__c`.

## Outputs

This method returns the working Map<String, Object>, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)  
[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)  
[ccrz\\_\\_E\\_OrderItemGroup\\_\\_c](#)  
[ccrz.ccApiAddress.createAddress](#)  
[ccrz.ccApiAddress.fetch](#)  
[Inserting and Updating Records](#)

## ccrz.ccLogicCartPlaceAsync.performDMLOnOrderItems

Calls inner methods for performing the DML `insert` operation for the order's major and minor line items, and then evaluates whether the cart has a coupon applied. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> performDMLOnOrderItems (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.processOrderGroups` method completes.

### Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

#### `ccrz.ccApiCart.HAS_COUPON_CODE`

Boolean

Value	Usage
<code>true</code>	The cart has a coupon applied.
<code>false</code>	The cart doesn't have any coupon applied.

#### `ccrz.ccLogicCartPlaceAsync.COUPON_TO_UPDATE`

`List<ccrz__E_CartCoupon__c>` that contains a coupon associated with the cart.



**Note:** Although this input is a list, this method evaluates only the first record in the list.

### Logic Performed

This method completes the following actions:

1. Calls the `ccrz.ccLogicCartPlaceAsync.insertMajorItems` inner method.
2. Calls the `ccrz.ccLogicCartPlaceAsync.insertMinorOrderItems` inner method.
3. Evaluates the `ccrz.ccApiCart.HAS_COUPON_CODE` input, and calls the `ccrz.ccApiCoupon.updateOnUse` method for updating the coupon's usage.



## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCoupon.COUPONS_UPDATED`

List<`ccrz__E_Coupon__c`> of the updated coupon records that the `ccrz.ccApiCoupon.updateOnUse` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)

[ccrz.ccApiCoupon.updateOnUse](#)

## `ccrz.ccLogicCartPlaceAsync.insertMajorItems`

Sets the `ccrz__Order__c` and `ccrz__RequestedDate__c` fields on each `ccrz__E_OrderItem__c` record that represents a major line item in the order. Then, the method performs the DML `insert` operation on the `ccrz__E_OrderItem__c` records. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> insertMajorItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlaceAsync.performDMLOnOrderItems` method.

This method requires the following keys in the working map:

### `ccrz.ccLogicCartPlaceAsync.MAJOR_ORDER_ITEMS`

Map<String, `ccrz__E_OrderItem__c`>, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is the `ccrz__E_OrderItem__c` to create for the corresponding major line item in the order.

### `ccrz.ccLogicCartPlaceAsync.ORDER_HEADER`

`ccrz__E_Order__c` record that represents the order to create.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartPlaceAsync.insertMinorOrderItems

Sets the `ccrz__Order__c`, `ccrz__RequestedDate__c`, and `ccrz__ParentOrderItem__c` fields on each `ccrz__E_OrderItem__c` record that represents a minor line item in the order. Then, the method performs the DML `insert` operation on the `ccrz__E_OrderItem__c` records. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> insertMinorOrderItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlace.performDMLOnOrderItems` method.

This method requires the following keys in the working map:

### **ccrz.ccLogicCartPlaceAsync.MAJOR\_ORDER\_ITEMS**

`Map<String, ccrz__E_OrderItem__c>`, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is the `ccrz__E_OrderItem__c` to create for the corresponding major line item in the order.

### **ccrz.ccLogicCartPlaceAsync.MINOR\_ORDER\_ITEMS**

`Map<String, List<ccrz__E_OrderItem__c>>`, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is a `List<ccrz__E_OrderItem__c>` to create as minor line items for the `ccrz__E_OrderItem__c` that corresponds to the major line item from the cart.

### **ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccLogicCartPlaceAsync.INVOICE\_ENABLED**

Boolean

Value	Usage
<i>true</i>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>true</code> .
<i>false</i>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>false</code> or <code>null</code> .

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.cc\_hk\_Invoice.PARAM\_ORDER\_CHILD\_ITEMS**

List<[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)> of the inserted minor line items.



**Note:** This output is returned only when the input data specifies `ccrz.ccLogicCartPlaceAsync.INVOICE_ENABLED => true`.

SEE ALSO:

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartPlaceAsync.enqueueNext

Determines whether more major line item `ccrz__E_CartItem__c` records remain to be processed, and calls `ccrz.ccLogicCartPlaceAsync.enqueuePlace` for queueing the next job. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> enqueueNext (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.performDMLOnOrderItems` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CARTITEMLASTID**

String that specifies the Salesforce ID of the final major cart item that was processed during the current recursive job.

 **Note:** If this input isn't specified, this method doesn't execute correctly and can return an error.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to convert.

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

## Logic Performed

This method completes the following actions:

1. Reads `ccrz.ccApiCart.CARTITEMLASTID` and determines where in the list of cart items the current recursive job finished processing.
2. Executes a direct query of `ccrz__E_CartItem__c` records for the source `ccrz__E_Cart__c` to determine whether more `ccrz__E_CartItem__c` records remain to be processed:
  - If more `ccrz__E_CartItem__c` records remain, this method sets the `ccrz.ccApiCart.NEXT_PLACE_JOB_ENQUEUED` output to *true*.
  - Otherwise, this method sets the `ccrz.ccApiCart.NEXT_PLACE_JOB_ENQUEUED` output to *false*.
3. Sets `ccrz.ccApiCart.CARTITEMLASTID` for the next recursive job that `ccrz.ccLogicCartPlaceAsync.enqueuePlace` executes.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApiCart.NEXT\_PLACE\_JOB\_ENQUEUED**

Boolean

Value	Usage
true	Another recursive <code>ccrz.ccApiCart.placeAsync</code> call is queued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
false	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccLogicCartPlaceAsync.enqueuePlace](#)

## ccrz.ccLogicCartPlaceAsync.processOrderTerms

Creates and inserts `ccrz__E_OrderTerm__c` records for the order. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processOrderTerms (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.enqueueNext` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiCart.NEXT\_PLACE\_JOB\_ENQUEUED**  
Boolean that must be *false* for this method to execute.

Value	Usage
<i>true</i>	Another recursive <code>ccrz.ccApiCart.placeAsync</code> call is enqueued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
<i>false</i>	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

**ccrz.ccLogicCartPlaceAsync.CART\_HEADER**  
`ccrz__E_Cart__c` record that represents the source cart.

**ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**  
`ccrz__E_Order__c` record that represents the order to create.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccLogicCartPlaceAsync.INVOICE\_ENABLED**  
Boolean

Value	Usage
<i>true</i>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <i>true</i> .
<i>false</i>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <i>false</i> or null.

## Logic Performed

This method completes the following actions:

1. Queries the `ccrz__E_Term__c` records for the storefront.
2. Creates `ccrz__E_OrderTerm__c` records for the order, based on the following criteria:
  - When the source `ccrz__E_Cart__c` record's `ccrz__ShipMethod__c` field isn't null, this method creates an `ccrz__E_OrderTerm__c` that references a `ccrz__E_Term__c` record where `ccrz__Type__c` is Shipping.
  - When the source `ccrz__E_Cart__c` record's `ccrz__TotalAmount__c` field is greater than zero, this method creates an `ccrz__E_OrderTerm__c` that references a `ccrz__E_Term__c` record where `ccrz__Type__c` is Billing.
3. Performs the DML `insert` operation on the `ccrz__E_OrderTerm__c` records.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.cc_hk_Invoice.PARAM_ORDER_TERMS`

`List<ccrz__E_OrderTerm__c>` of the inserted order terms.



**Note:** This output is returned only when the input data specifies `ccrz.ccLogicCartPlaceAsync.INVOICE_ENABLED => true`.

SEE ALSO:

[ccrz\\_\\_E\\_OrderTerm\\_\\_c](#)

[ccrz\\_\\_E\\_Term\\_\\_c](#)

[Inserting and Updating Records](#)

## `ccrz.ccLogicCartPlaceAsync.closeCart`

Updates fields on the source `ccrz__E_Cart__c` record to indicate that the cart is closed and a corresponding order was created. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> closeCart (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.processOrderTerms` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiCart.NEXT\_PLACE\_JOB\_ENQUEUED**  
Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Another recursive <code>ccrz.ccApiCart.placeAsync</code> call is enqueued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
<code>false</code>	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

**ccrz.ccLogicCartPlaceAsync.CART\_HEADER**  
`ccrz__E_Cart__c` record that represents the source cart.

## Logic Performed

This method completes the following actions:

1. Sets the following fields on the source `ccrz__E_Cart__c` record:

Field	Value
<code>ccrz__ActiveCart__c</code>	<code>false</code>
<code>ccrz__CartStatus__c</code>	<code>Closed</code>

2. Performs the DML `update` operation on the `ccrz__E_Cart__c` record.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)  
[Inserting and Updating Records](#)

## ccrz.ccLogicCartPlaceAsync.createStoredPayments

Evaluates whether to create a stored payment from the input payment data. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> createStoredPayments (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.closeCart` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiCart.NEXT\_PLACE\_JOB\_ENQUEUED**

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Another recursive <code>ccrz.ccApiCart.placeAsync</code> call is enqueued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
<code>false</code>	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

### **ccrz.ccApiCart.PAYMENTDATA**

JSON-formatted string that describes payment information passed to the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA` input.

### **ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

### **ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create.

## Logic Performed

This method completes the following actions:

1. Checks the value of the My Wallet storefront configuration module's Enabled setting (`wlt.enabled`). If the configuration is `true`, execution continues.
2. Parses `ccrz.ccApiCart.PAYMENTDATA` to determine whether the buyer chose to save the payment information as a stored payment. If so, execution continues.
3. Calls the `ccrz.cc_hk_Payment.createStoredPayment` extension point method for creating a `ccrz__E_StoredPayment__c` record from the payment data.



## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.createStoredPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

SEE ALSO:

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

[ccrz.cc\\_hk\\_Payment.createStoredPayment](#)

[ccrz.cc\\_hk\\_Payment.processPayment](#)

[Salesforce Help Configure Payment Options Available on the Checkout Page](#)

## ccrz.ccLogicCartPlaceAsync.processSubscriptions

Creates `ccrz__E_Subscription__c` records for any `ccrz__E_SubProdTerm__c` records associated with the order. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processSubscriptions(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.createStoredPayments` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiCart.NEXT\_PLACE\_JOB\_ENQUEUED**

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Another recursive <code>ccrz.ccApiCart.placeAsync</code> call is enqueued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
<code>false</code>	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

**ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.createStoredPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

**ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create.

**'hasSubProdTerm'**

Boolean

Value	Usage
<i>true</i>	The source <code>ccrz__E_Cart__c</code> included a <code>ccrz__E_CartItem__c</code> with an associated <code>ccrz__E_SubProdTerm__c</code> record.
<i>false</i>	The source <code>ccrz__E_Cart__c</code> didn't have subscription data. This method exits without evaluating subscriptions.

## Logic Performed

This method completes the following actions:

1. Evaluates the 'hasSubProdTerm' input.
2. Calls the `ccrz.cc_hk_Subscriptions.onOrder` extension point method for creating the order's `ccrz__E_Subscription__c` records.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccLogicCartPlaceAsync.SUB\_RESULT**

`Map<String, Object>` that the `ccrz.cc_hk_Subscriptions.onOrder` method returned.

## ccrz.ccLogicCartPlaceAsync.processTransactionPayments

Evaluates whether to call the `ccrz.cc_hk_Order.createTransaction` extension point method, and then calls `ccrz.ccPaymentProcessor` methods for processing the payment. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processTransactionPayments (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.processSubscriptions` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiCart.NEXT\_PLACE\_JOB\_ENQUEUED**

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Another recursive <code>ccrz.ccApiCart.placeAsync</code> call is enqueued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
<code>false</code>	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

### **ccrz.ccApiCart.PAYMENTDATA**

JSON-formatted string that describes payment information passed to the `ccrz.cc_hk_Payment.processPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA` input.

### **ccrz.ccApiCart.PAYMENTRESULT**

`Map<String, Object>` that represents the output from the `ccrz.cc_hk_Payment.processPayment` extension point method.

### **ccrz.ccApiCart.TRANSPAYMENTDATA**

JSON-formatted string that describes payment information returned in the `ccrz.cc_hk_Payment.createStoredPayment` extension point method's `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` output.

### **ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to create.

### **ccrz.ccLogicCartPlaceAsync.SUB\_RESULT**

`Map<String, Object>` that the `ccrz.cc_hk_Subscriptions.onOrder` method returned.

## Logic Performed

This method completes the following actions:

1. Verifies that `ccrz.ccApiCart.PAYMENTDATA` and `ccrz.ccApiCart.TRANSPAYMENTDATA` exist for the order.
2. Calls the `ccrz.cc_hk_Order.createTransaction` extension point method for creating the order's transaction payment details.
3. Calls the `ccrz.ccPaymentProcessor.process` utility method for processing the transaction payment.
4. Calls the `ccrz.ccPaymentProcessor.postProcess` utility method for handling the payment's success or failure.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccLogicCartPlaceAsync.TRANSACTION\_RESULT**

Map<String, Object> that the `ccrz.cc_hk_Order.createTransaction` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

[ccrz.cc\\_hk\\_Order.createTransaction](#)

[ccrz.ccPaymentProcessor.process](#)

[ccrz.ccPaymentProcessor.postProcess](#)

**ccrz.ccLogicCartPlaceAsync.changeOrderStatus**

Updates fields on the created `ccrz__E_Order__c` record to indicate that the order was placed. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

**Signature**

```
global virtual Map<String, Object> changeOrderStatus (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> after the `ccrz.ccLogicCartPlaceAsync.processTransactionPayments` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiCart.NEXT\_PLACE\_JOB\_ENQUEUED**

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Another recursive <code>ccrz.ccApiCart.placeAsync</code> call is enqueued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
<code>false</code>	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

**ccrz.ccLogicCartPlaceAsync.CART\_HEADER**

`ccrz__E_Cart__c` record that represents the source cart.

**ccrz.ccLogicCartPlaceAsync.ORDER\_HEADER**

`ccrz__E_Order__c` record that represents the order to place.

**Inputs (Optional)**

If the working map includes the following key, this method evaluates it:

**ccrz.ccApiOrder.PARAM\_ORDER\_STATUS**

String that specifies a status to set on the order when it's created.

**Logic Performed**

This method completes the following actions:

1. Sets the `ccrz__OrderStatus__c` field on the `ccrz__E_Order__c` record, in the following order of precedence:
  - a. The value of the `ccrz.ccApiOrder.PARAM_ORDER_STATUS` input, if specified.
  - b. `On Hold` if the cart or order indicates a tax exemption.
  - c. `Order Submitted` for all other cases.
2. Performs the DML `update` operation on the `ccrz__E_Order__c` record.

**Outputs**

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[Inserting and Updating Records](#)

**ccrz.ccLogicCartPlaceAsync.processInvoices**

Calls the `ccrz.cc_hk_Invoice.onOrder` extension point method for creating invoice for the order. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

**Signature**

```
global virtual Map<String, Object> processInvoices (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.changeOrderStatus` method completes.

This method requires the following keys in the working map:

### `ccrz.ccApiCart.NEXT_PLACE_JOB_ENQUEUED`

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Another recursive <code>ccrz.ccApiCart.placeAsync</code> call is enqueued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
<code>false</code>	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

### `ccrz.ccLogicCartPlaceAsync.CART_HEADER`

`ccrz__E_Cart__c` record that represents the source cart.

### `ccrz.ccLogicCartPlaceAsync.INVOICE_ENABLED`

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>true</code> .
<code>false</code>	The Invoice storefront configuration module's Enabled setting ( <code>inv.enabled</code> ) is <code>false</code> or null.

### `ccrz.ccLogicCartPlaceAsync.MAJOR_ORDER_ITEMS`

`Map<String, ccrz__E_OrderItem__c>`, where:

- Each key is the Salesforce ID of a source `ccrz__E_CartItem__c` record that represents a major line item in the cart.
- Each value is the `ccrz__E_OrderItem__c` to create for the corresponding major line item in the order.

### `ccrz.ccLogicCartPlaceAsync.ORDER_HEADER`

`ccrz__E_Order__c` record that represents the order to place.

### `ccrz.cc_hk_Invoice.PARAM_CART_ITEMS`

`List<ccrz__E_CartItem__c>` of the original cart items included in the `ccrz.ccLogicCartPlace.CART_ITEM_FETCH_RESPONSE` input.

### `ccrz.cc_hk_Invoice.PARAM_ORDER_CHILD_ITEMS`

`List<ccrz__E_OrderItem__c>` of the order's minor line items.

### `ccrz.cc_hk_Invoice.PARAM_ORDER_TERMS`

`List<ccrz__E_OrderTerm__c>` of the inserted order terms.

## Logic Performed

This method completes the following actions:

1. Verifies that `ccrz.ccLogicCartPlaceAsync.INVOICE_ENABLED` is `true`. Otherwise, the method exits.

2. Passes the other inputs to the `ccrz.cc_hk_Invoice.onOrder` extension point method for creating the `ccrz__E_Invoice__c` record for the order.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Invoice\\_\\_c](#)

[Salesforce Help Generate an Invoice for a Buyer's Order](#)

## ccrz.ccLogicCartPlaceAsync.sendOrderConfirmation

Calls the `System.enqueueJob` method for adding the queueable `ccrz.cc_util_OrderConfirmationEmail.ccUtilOrderConfirmationEmail` class to the asynchronous Apex job queue for sending an order confirmation email. This method executes only during the final recursive `ccrz.ccApiCart.placeAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> sendOrderConfirmation (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPlaceAsync.processInvoices` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiCart.NEXT\_PLACE\_JOB\_ENQUEUED**

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Another recursive <code>ccrz.ccApiCart.placeAsync</code> call is enqueued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
<code>false</code>	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

**ccrz.ccApiCart.ORDER\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Order__c` record.

**Inputs (Optional)**

If the working map includes the following key, this method evaluates it:

**ccrz.ccLogicCartPlace.EMAIL\_TEMPLATE\_CONFIRM**

String that specifies the name of the email template to use for the order confirmation email. The default template is `CC_Order_AsyncConfirm`. Optionally, specify a custom email template instead.

**Logic Performed**

This method first verifies that the Send Confirmation Email storefront setting (`orderFromEmail__c`) is enabled before enqueueing the class that sends the order confirmation email.

**Outputs**

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[enqueueJob \(queueableObj\)](#)

[Queueable Apex](#)

[Salesforce Help: Show Order Confirmation Details and Send Order Emails/](#)

**ccrz.ccLogicCartPlaceAsync.updateOrderOnError**

Updates the status of the `ccrz__E_Order__c` record and deletes the order's billing or shipping `ccrz__E_ContactAddr__c` records when an exception occurs during asynchronous order placement. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.placeAsync` method calls.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

**Signature**

```
global virtual Map<String, Object> updateOrderOnError (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlaceAsync.executeAsyncOrder` method.

This method requires the following keys in the working map:



**ccrz.ccApiCart.ORDER\_ID**


String that specifies the Salesforce ID of the `ccrz__E_Order__c` record.

**Inputs (Optional)**

If the working map includes the following key, this method evaluates it:

**ccrz.ccApiCart.PRICING\_FOR\_ORDER\_FAILED**

Boolean that must be `true` for this method to execute.

Value	Usage
<code>true</code>	<p>An error occurred when pricing the cart during the previous iteration of the recursive, asynchronous call. The logic chain returns a <code>ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException</code>.</p> <p> <b>Note:</b> This input is used only when the <code>ccrz.ccApiCart.ALLOW_PRICING_ON_ASYNC_ORDER</code> input is <code>true</code> and the cart was repriced because its price data wasn't current.</p>
<code>false</code>	No error occurred.

**Logic Performed**

This method completes the following actions:

1. Verifies that the working map includes the `ccrz.ccApiCart.PRICING_FOR_ORDER_FAILED` input with a value of `true`.
2. Performs the DML `update` operation on the `ccrz__E_Order__c` and sets the `ccrz__OrderStatus__c` field to `Error`.
3. Performs the DML `delete` operation on the `ccrz__E_ContactAddr__c` records that the `ccrz__E_Order__c` record's `ccrz__BillTo__c` and `ccrz__ShipTo__c` fields reference.
4. Calls the `ccrz.ccLogicCartPlaceAsync.sendOrderFail` inner method for sending an order failure email.

**Outputs**

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[Inserting and Updating Records](#)

[Deleting Records](#)

**ccrz.ccLogicCartPlaceAsync.sendOrderFail**

Calls the `System.enqueueJob` method for adding the queueable

`ccrz.cc_util_OrderConfirmationEmail.ccUtilOrderConfirmationEmail` class to the asynchronous Apex job queue for sending an order failure email.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> sendOrderFail (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPlaceAsync.updateOrderOnError` method.

This method requires the following keys in the working map:

### **ccrz.ccApiCart.ORDER\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Order__c` record.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### **ccrz.ccLogicCartPlace.EMAIL\_TEMPLATE\_FAIL**

String that specifies the name of the email template to use for the order failure email. The default template is `CC_Order_AsyncFail`. Optionally, specify a custom email template instead.

## Logic Performed

This method first verifies that the Send Confirmation Email storefront setting (`orderFromEmail__c`) is enabled before enqueueing the class that sends the order failure email.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[enqueueJob \(queueableObj\)](#)

[Queueable Apex](#)

[Salesforce Help: Show Order Confirmation Details and Send Order Emails/](#)

## ccrz.ccLogicCartPlaceAsync.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys. This method executes for every `ccrz.ccApiCart.placeAsync` call, regardless of whether it's the initial or a recursive call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the appropriate method completes, depending on which portion of the logic chain is currently executing:

### Initial synchronous call

`ccrz.ccLogicCartPlaceAsync.enqueuePrice` or `ccrz.ccLogicCartPlaceAsync.enqueuePlace` during `ccrz.ccLogicCartPlaceAsync.prepareAsyncOrder`

### Recursive, asynchronous calls

`ccrz.ccLogicCartPlaceAsync.enqueueNext` during `ccrz.ccLogicCartPlaceAsync.executeAsyncOrder`

### Final recursive call

`ccrz.ccLogicCartPlaceAsync.sendOrderConfirmation` during `ccrz.ccLogicCartPlaceAsync.executeAsyncOrder`

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`:

- `ccrz.ccApiCart.PAYMENTDATA`
- `ccrz.ccApiCart.PAYMENTRESULT`
- `ccrz.ccApiCart.TRANSPAYMENTDATA`
- `ccrz.ccApiOrder.PARAM_ORDER_STATUS`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicCartPlaceAsync.CART_HEADER`
- `ccrz.ccLogicCartPlaceAsync.INVOICE_ENABLED`
- `ccrz.ccLogicCartPlaceAsync.MAJOR_ORDER_ITEMS`
- `ccrz.ccLogicCartPlaceAsync.MINOR_ORDER_ITEMS`
- `ccrz.ccLogicCartPlaceAsync.ORDER_BILLING_ADDRESS`
- `ccrz.ccLogicCartPlaceAsync.ORDER_HEADER`
- `ccrz.ccLogicCartPlaceAsync.ORDER_SHIPPING_ADDRESS`
- `ccrz.cc_hk_Invoice.PARAM_CART_ITEMS`
- `ccrz.cc_hk_Invoice.PARAM_ORDER_CHILD_ITEMS`
- `ccrz.cc_hk_Invoice.PARAM_ORDER_TERMS`

## ccrz.ccLogicCartPlaceAsync.handleCartPlaceException

Catches a `ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException` that occurs during the logic chain, and logs the exception message.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> handleCartPlaceException(Map<String, Object>)
```

### Inputs (Required)

`Map<String, Object>` that can include the following keys:

#### 'CartPlaceException'

`ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException` that occurred during the `ccrz.ccLogicCartPlaceAsync.process` method or its inner methods. The

`ccrz.ccLogicCartPlaceAsync.process` method rethrows the exception after it's caught, which ends processing and rolls back transactions.

### Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

## ccrz.ccLogicCartPrice

Define the business logic for updating the price information for the items in a specified cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**!** **Important:** In B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12), this logic service provider class contains some differences from older versions. Pay close attention to the compatibility information listed for each inner method.

## Global API Caller

`ccrz.ccApiCart.price`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [checkCartForAsyncPricing](#)

Evaluates the cart, storefront configuration settings, and input keys to determine whether to price the cart synchronously or asynchronously.

#### [retrieveCartData](#)

Calls the `ccrz.ccApiCart.fetch` method for querying the cart to price.

#### [validateLastPriceDate](#)

Evaluates whether the `ccrz__RepricedDate__c` field on the `ccrz__E_Cart__c` record is considered current, depending on the value of the Optimize Cart Repricing storefront configuration.

#### [processCart](#)

Evaluates the list of `ccrz__E_CartItem__c` records in the fetched cart, and sets null values for discount fields. Optionally, this method also consumes the result of the `ccrz.cc_api_CartExtension.preprocess` extension point method.

#### [prepareCartItems](#)

Identifies `ccrz__E_CartItem__c` records to pass to or exclude from the subsequent pricing methods.

#### [fetchPricingData](#)

Calls the `ccrz.ccApiProduct.fetch` method and returns product pricing data for the cart items to reprice. If the fetched product data includes attributes, this method also calls the `ccrz.ccApiAttribute.fetch` method.

#### [applyPricing](#)

Loops through the cart items to reprice, and calls the appropriate inner method for setting each cart item's price. This method also ensures that major and minor line items remain associated with each other.

#### [priceltem](#)

Begins the For loop within the `ccrz.ccLogicCartPrice.applyPricing` method by evaluating the current cart item and determining the appropriate pricing logic.

#### [evaluateSubProdtermPricing](#)

Parses the subscription term for a cart item, and sets the item's corresponding price fields.

[applySubscriptionPercentages](#)

Adjusts the price for a cart item that's a minor line item for a kit or dynamic kit that defines a subscription term. This method applies the parent subscription's upfront and recurring percentage discounts to the child cart item, which previously had only standard pricing logic applied.

[priceAttributeItem](#)

Parses JSON attribute pricing data for a cart item, and sets the item's corresponding price fields.

[checkCartItemValidity](#)

Continues the For loop within the `ccrz.ccLogicCartPrice.applyPricing` method by evaluating validation criteria and updating the cart item's status.

[evaluateComplexProducts](#)

Evaluates cart items that are included in complex products and sets the status of each cart item. A *complex product* can be a bundle, kit, dynamic kit, assembly product, or attribute-based product.

[performExternalPricing](#)

Provides a placeholder for pricing a cart from an external source. By default, this method is empty and performs no default action.

[performDML](#)

Removes cart items, combines externally priced cart items with other cart items, and finally executes the DML `upsert` operation for the updated cart item records.

[applyCoupons](#)

Creates a cart item for a coupon, and applies the coupon's discount to the cart.

[refetch](#)

Call the `ccrz.ccApiCart.fetch` method for fetching the repriced and updated cart and cart items.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCartPrice.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.price` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartPrice` class in the following order:

1. `processInput`
2. `checkCartForAsyncPricing`



### Note:

- This method executes only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12).
- If the `ccrz.ccApiCart.PRICE_ASYNC` key is `false` after this method completes, the `ccrz.ccApiCart.priceAsync` global method executes instead. The rest of the methods in this logic service provider class are skipped.

3. `retrieveCartData`
4. `validateLastPriceDate`
5. `processCart`
6. `prepareCartItems`
7. `fetchPricingData`
8. `applyPricing`, which calls the following inner methods:
  - a. `priceItem` for each cart item, which calls into the appropriate method for pricing different product scenarios:
    - `evaluateSubProdTermPricing`
    - `applySubscriptionPercentages`
    - `priceAttributeItem`
  - b. `checkCartItemValidity` for each cart item
  - c. `evaluateComplexProducts`
9. `performExternalPricing`
10. `performDML`, which calls into `applyCoupons`
11. `refetch`
12. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartPrice.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```


### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartPrice.process` method.

This method checks that the input map specifies the following keys, and returns a `ccrz.ccApi.MissingInputException` if any key is missing:

#### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

#### **ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted cart ID to price.

#### **ccrz.ccApiCart.CART\_ID**

String that specifies the cart ID to price.

 **Note:** Include `ccrz.ccApiCart.CART_ENCID` or `ccrz.ccApiCart.CART_ID`, but not both.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:



**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The [ccrz.ccLogicCartPrice.prepReturn method](#) removes this intermediate key from the working map.

**ccrz.ccLogicCartPrice.checkCartForAsyncPricing**

Evaluates the cart, storefront configuration settings, and input keys to determine whether to price the cart synchronously or asynchronously.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

**Signature**

```
global virtual Map<String, Object> checkCartForAsyncPricing (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the Map<String, Object> after the [ccrz.ccLogicCartPrice.processInput](#) method completes.

This method requires either of the following keys in the working map:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted cart ID to price.

**ccrz.ccApiCart.CART\_ID**

String that specifies the cart ID to price.

**Inputs (Optional)**

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiCart.FORCE\_PRICE\_SYNC**

Boolean

Value	Usage
<i>true</i>	Execute pricing synchronously, regardless of the value of the <a href="#">ccrz.ccApiCart.PRICE_ASYNC</a> input.
<i>false</i> (default)	Default logic determines whether to price the cart synchronously or asynchronously.

**ccrz.ccApiCart.PARTIAL\_CART\_UPDATE**

Boolean

Value	Usage
<i>true</i>	Apply pricing, entitlements, and quantity updates only to a certain subset of cart items. The <code>ccrz.ccLogicCartPrice.retrieveCartData</code> method defines the business logic for the partial update.
<i>false</i> (default)	Execute default pricing logic.

**ccrz.ccApiCart.PRICE\_ASYNC**

Boolean

Value	Usage
<i>true</i>	Execute pricing asynchronously by calling the <code>ccrz.ccApiCart.priceAsync</code> method.
<i>false</i> (default)	Execute pricing synchronously.

**ccrz.ccApiCart.SYNC\_MAX\_CART\_SIZE**

Integer that specifies the maximum number of items that must be in a cart for synchronous processing. This value overrides the value of the Asynchronous Processing Threshold storefront configuration setting in the Cart Scale module (`cs.asyncthreshold`).

**Logic Performed**

This method completes the following actions:

1. Evaluates whether any of the following criteria for *synchronous* cart pricing are satisfied:

- The `ccrz.ccApiCart.FORCE_PRICE_SYNC` input key is *true*.
- The `ccrz.ccApiCart.PARTIAL_CART_UPDATE` input key is *true*.
- The buyer is a guest.
- The Cart Scale storefront configuration module's Enable Cart Scale setting (`cs.enable`) is *false* or null.

If any of these criteria are satisfied, this method sets `ccrz.ccApiCart.PRICE_ASYNC` to *false*, and the method exits. Otherwise, execution continues.

2. Evaluates whether the `ccrz.ccApiCart.PRICE_ASYNC` input key already specifies a value.

3. Completes the following actions when the Cart Scale storefront configuration module's Enable Cart Scale setting (`cs.enable`) is *true*:

- Evaluates whether the `ccrz.ccApiCart.SYNC_MAX_CART_SIZE` input key already specifies a value. If so, the method uses that input's value. Otherwise, this method finds the value of the Cart Scale storefront configuration module's Asynchronous Processing Threshold setting (`cs.asyncthreshold`).
- Passes `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID` as input for the `ccrz.ccApiCart.fetch` method.
- Finds the value of the fetched `ccrz__E_Cart__c` record's `ccrz__NumberOfAllCartItems__c` field.
- Compares the total number of cart items with the `ccrz.ccApiCart.SYNC_MAX_CART_SIZE` input value or Asynchronous Processing Threshold configuration value, whichever is used. When the number of cart items exceeds the processing threshold, the cart is priced asynchronously.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**`ccrz.ccApiCart.PRICE_ASYNC`**

Boolean

Value	Usage
<code>true</code>	The criteria determined that the cart is priced asynchronously. This logic service provider skips the remaining methods in its logic chain, and calls <code>ccrz.ccApiCart.priceAsync</code> instead.
<code>false</code>	The criteria determined that the cart is priced synchronously. This logic service provider's execution continues.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz.ccApiCart.fetch](#)

## `ccrz.ccLogicCartPrice.retrieveCartData`

Calls the `ccrz.ccApiCart.fetch` method for querying the cart to price.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> retrieveCartData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the appropriate method completes, depending on which version of the managed package you're using:

**B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12)**

[ccrz.ccLogicCartPrice.checkCartForAsyncPricing](#)

**B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11) or earlier**`ccrz.ccLogicCartPrice.processInput`

This method requires either of the following keys in the working map:

**`ccrz.ccApiCart.CART_ENCID`**

String that specifies the encrypted cart ID to price.

**`ccrz.ccApiCart.CART_ID`**

String that specifies the cart ID to price.

 **Note:** If neither key is specified, this method returns `ccrz.ccApiCart.CartPriceException`.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**`ccrz.ccApiCart.PARTIAL_CART_UPDATE`**

Boolean


Value	Usage
<code>true</code>	Apply pricing, entitlements, and quantity updates only to a certain subset of cart items.
<code>false</code> (default)	Execute default pricing logic.

## Logic Performed

This method completes the following actions, depending on which version of the managed package you're using:

**B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12)**

1. Calls the `ccrz.ccApiCart.fetch` method for returning the cart to price.
2. Calls the `ccrz.ccApiCart.fetchCartItems` method for returning the cart's items.

 **Note:** If the working map's `ccrz.ccApiCart.PARTIAL_CART_UPDATE` input is `true`, the `ccrz.ccApiCart.fetchCartItems` method call is adjusted in the following ways:

- Queries only a subset of `ccrz__E_CartItem__c` records, which the `ccrz.ccApiCart.CARTITEM_IDS` input specifies.
- Passes a value of `true` for the `ccrz.ccApiCart.CARTITEM_INCLUDE_CHILDREN` input.

**B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11) or earlier**

Calls only the `ccrz.ccApiCart.fetch` method for returning the cart to price.

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiCart.fetch` outputs, and sets the following intermediate keys:

**'cartItemList'**

List<`ccrz__E_CartItem__c`> of all the cart's items.

**'cartItemsToUpdate'**

Map<String, Map<String, Object>> of the `ccrz__E_CartItem__c` records to reprice. For example, when a buyer updates quantities of items in the cart, this list specifies the cart items that have changed. The working map's `ccrz.ccApiCart.CILIST` specified this map, which is keyed by cart item ID, and each nested Map<String, Object> defines the following subkeys:

**ccrz.ccApiCart.CIQTY**

Integer that specifies the item's quantity.

**ccrz.ccApiCart.CICMTS**

String that specifies the item's comments.

**'cartToPrice'**

`ccrz__E_Cart__c` that represents the cart to price.

**'skpPreProc'**

Boolean

Value	Usage
<i>true</i>	The working map's <code>ccrz.ccApiCart.CIPREPROC</code> key specified a value of <i>false</i> to skip calling the <code>ccrz.cc_api_CartExtension.preprocess</code> extension point method.
<i>false</i> (default)	The working map's <code>ccrz.ccApiCart.CIPREPROC</code> key specified a value of <i>true</i> to call the <code>ccrz.cc_api_CartExtension.preprocess</code> extension point method.

## SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[ccrz.ccApiCart.fetchCartItems](#)

**ccrz.ccLogicCartPrice.validateLastPriceDate**

Evaluates whether the `ccrz__RepricedDate__c` field on the `ccrz__E_Cart__c` record is considered current, depending on the value of the Optimize Cart Repricing storefront configuration.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

## Signature

```
global virtual Map<String, Object> validateLastPriceDate (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartPrice.retrieveCartData` method.

This method requires the following key in the working map:

'**cartToPrice**'

`ccrz__E_Cart__c` that represents the cart to evaluate.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiCart.SKIP\_CART\_PRICING**

Boolean

Value	Usage
<i>true</i>	Skip the logic in this method, and immediately return the working <code>Map&lt;String, Object&gt;</code> .
<i>false</i> (default)	Execute the logic in this method.

## Logic Performed

This method completes the following actions:

1. Evaluates whether input data already specifies a value for `ccrz.ccApiCart.SKIP_CART_PRICING`. Otherwise, the method continues.
2. Finds the value of the Cart storefront configuration module's Optimize Cart Repricing setting (`c.optCartReprice`).
3. Evaluates whether any of the following criteria are satisfied, and if so, sets a value for `ccrz.ccApiCart.SKIP_CART_PRICING`:
  - If the `ccrz.ccApiCart.HAS_COUPON_CODE` input is *true*, the method sets `ccrz.ccApiCart.SKIP_CART_PRICING` to *false*.
  - If the cart's `ccrz__RepricedDate__c` date matches the current date, the method sets `ccrz.ccApiCart.SKIP_CART_PRICING` to *true*.
  - If the cart's `ccrz__RepricedDate__c` is more recent than the Valid Prices Date storefront setting, the method sets `ccrz.ccApiCart.SKIP_CART_PRICING` to *true*.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiCart.SKIP\_CART\_PRICING**

Boolean

Value	Usage
true	Skip the following methods: <ul style="list-style-type: none"> <li>• <code>processCart</code></li> <li>• <code>prepareCartItems</code></li> <li>• <code>fetchPricingData</code></li> <li>• <code>applyPricing</code></li> <li>• <code>performDML</code></li> </ul> This method returns <code>true</code> when either or both of the following conditions are satisfied:
false	Execute the next method in the process chain.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[Salesforce Help Optimize How Frequently the Shopping Cart Page Calls Price Logic](#)

## ccrz.ccLogicCartPrice.processCart

Evaluates the list of `ccrz__E_CartItem__c` records in the fetched cart, and sets null values for discount fields. Optionally, this method also consumes the result of the `ccrz.cc_api_CartExtension.preprocess` extension point method.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processCart (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartPrice.validateLastPriceDate` method.

This method requires the following keys in the working map:

**'cartItemList'**

List<ccrz\_\_E\_CartItem\_\_c> of all the cart's items.

**'cartToPrice'**

ccrz\_\_E\_Cart\_\_c that represents the cart to evaluate.

**ccrz.ccApiCart.SKIP\_CART\_PRICING**

Boolean

Value	Usage
<i>true</i>	Skip this method.
<i>false</i>	Execute this method.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**'skpPreProc'**

Boolean

Value	Usage
<i>true</i>	Skip calling the <code>ccrz.cc_api_CartExtension.preprocess</code> extension point method.
<i>false</i> (default)	<p>Call the <code>ccrz.cc_api_CartExtension.preprocess</code> extension point method, and apply its return data to the working map:</p> <ul style="list-style-type: none"> <li>Use the cart specified by <code>ccrz.cc_api_CartExtension.PARAM_CART</code> for this method's 'cartToPrice'.</li> <li>Use the cart items specified by <code>ccrz.cc_api_CartExtension.PARAM_CARTITEMS</code> for this method's 'cartItemList'.</li> </ul>

## Outputs

This method returns the working Map<String, Object>, and sets the following fields on each ccrz\_\_E\_CartItem\_\_c record:

ccrz__E_CartItem__c Field	Value
ccrz__AbsoluteDiscount__c	null
ccrz__AdjustmentAmount__c	null
ccrz__PercentDiscount__c	null



ccrz__E_CartItem__c Field	Value
ccrz__PricingModifierTypes__c	null

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.cc\\_api\\_CartExtension.preprocess](#)

## ccrz.ccLogicCartPrice.prepareCartItems

Identifies `ccrz__E_CartItem__c` records to pass to or exclude from the subsequent pricing methods.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepareCartItems (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartPrice.processCart` method.

This method requires the following keys in the working map:

#### 'cartItemList'

List<`ccrz__E_CartItem__c`> of all the cart's items.

#### 'cartItemsToUpdate'

Map<String, Map<String, Object>> of the `ccrz__E_CartItem__c` records to reprice. For example, when a buyer updates quantities of items in the cart, this list specifies the cart items that have changed. The working map's `ccrz.ccApiCart.CILIST` specified this map, which is keyed by cart item ID, and each nested `Map<String, Object>` defines the following subkeys:

#### **ccrz.ccApiCart.CIQTY**

Integer that specifies the item's quantity.

#### **ccrz.ccApiCart.CICMTS**

String that specifies the item's comments.

**'cartToPrice'**

`ccrz__E_Cart__c` that represents the cart to evaluate.

**ccrz.ccApiCart.SKIP\_CART\_PRICING**

Boolean

Value	Usage
<code>true</code>	Skip this method.
<code>false</code>	Execute this method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**'attributes'**

`Set<String>` of attribute IDs, populated from each cart item's `ccrz__PrimaryAttr__c`, `ccrz__SecondaryAttr__c`, and `ccrz__TertiaryAttr__c` fields.


**'cartItemsToDelete'**

`List<ccrz__E_CartItem__c>` of the items to remove from the working map. By default, this method adds a cart item to this list when the item meets any of the following criteria:

- The cart item's quantity changed to `0`.
- The cart item's `ccrz__ProductType__c` field has the value `Coupon`.

**'cartItemsToPrice'**

`List<ccrz__E_CartItem__c>` of the items to pass to subsequent pricing methods. For each cart item, this method updates the values of the following fields:

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__Comments__c</code>	Value of the 'cartItemsToUpdate' input key's <code>ccrz.ccApiCart.CICMTS</code> subkey
<code>ccrz__ItemStatus__c</code>	Available
<code>ccrz__Quantity__c</code>	Value of the 'cartItemsToUpdate' input key's <code>ccrz.ccApiCart.CIQTY</code> subkey
<code>ccrz__Price__c</code>	-1  <b>Note:</b> This value is an initial value. The inner methods of <code>ccrz.ccLogicCartPrice.applyPricing</code> evaluate each cart item and update this value.

**'externalCartItems'**

`List<ccrz__E_CartItem__c>` of the items where `ccrz__PricingType__c` is `external`. These items are excluded from the subsequent pricing methods.

**'majorsWExtChild'**

List<String> of cart item IDs that are major line items with child cart items where `ccrz__PricingType__c` is external. These items are excluded from the subsequent pricing methods.

**'pricingInputs'**

Map<String, Object> of input data for a `ccrz.ccApiProduct.fetch` query, which the `ccrz.ccLogicCartPrice.fetchPricingData` method executes. This map includes:

- Available data from other return data keys—such as 'products', 'attributes', 'sellers', and 'spts'—which specify parameters for the fetch query.
- The key value `ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true`, which ensures that the fetch query returns `ccrz.ccApiProduct.PRODUCTPRICINGDATA` and `ccrz.ccApiAttribute.IDS`, if applicable.

**'products'**

Set<String> of product IDs, populated from each cart item's `ccrz__Product__c` field.

**'sellers'**

Set<String> of seller IDs, populated from each cart item's `ccrz__Seller__c` field.

**'spts'**

Set<String> of subscription term IDs, populated from each cart item's `ccrz__SubProdTerm__c` field.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccApiProduct.fetch](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz.ccLogicCartPrice.fetchPricingData

Calls the `ccrz.ccApiProduct.fetch` method and returns product pricing data for the cart items to reprice. If the fetched product data includes attributes, this method also calls the `ccrz.ccApiAttribute.fetch` method.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchPricingData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartPrice.prepareCartItems` method.

This method requires the following key in the working map:

### 'pricingInputs'

`Map<String, Object>` of input data for a `ccrz.ccApiProduct.fetch` query. This map includes:

- Products, attributes, sellers, and subscription terms parameters, if applicable.
- The key value `ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true`, which ensures that the fetch query returns `ccrz.ccApiProduct.PRODUCTPRICINGDATA` and `ccrz.ccApiAttribute.IDS`, if applicable.

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiProduct.fetch` outputs. In particular, the subsequent pricing methods evaluate the following output keys:

### `ccrz.ccApiAttribute.ATTRIBUTES`

`Map<String, Object>` that describes `ccrz__E_Attribute__c` records related to the queried products. The `ccrz.ccApiAttribute.fetch` method returns this map.

### `ccrz.ccApiProduct.PRODUCTPRICINGDATA`

`Map<String, Object>` of pricing data for all queried products. This map can contain seller, subscription term, attribute, and standard pricing information for the queried products.

SEE ALSO:

[ccrz.ccApiProduct.fetch](#)

[ccrz.ccApiAttribute.fetch](#)

## `ccrz.ccLogicCartPrice.applyPricing`

Loops through the cart items to reprice, and calls the appropriate inner method for setting each cart item's price. This method also ensures that major and minor line items remain associated with each other.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> applyPricing (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartPrice.fetchPricingData` method.

This method requires the following keys in the working map:

### 'cartItemsToPrice'

`Map<String, ccrz__E_CartItem__c>` of the items to price.

### ccrz.ccApiAttribute.ATTRIBUTES

`Map<String, Object>` that describes `ccrz__E_Attribute__c` records related to the queried products. The `ccrz.ccApiAttribute.fetch` method returns this map.

### ccrz.ccApiProduct.PRODUCTPRICINGDATA

`Map<String, Object>` of pricing data for all queried products. This map can contain seller, subscription term, attribute, and standard pricing information for the queried products.

## Logic Performed

This method completes the following actions:

1. For each cart item, the method creates an intermediate map and passes it to the `ccrz.ccLogicCartPrice.priceItem` inner method:

### pricingInputs

`Map<String, Object>` that contains the following keys that the `ccrz.ccLogicCartPrice.priceItem` inner method evaluates for an individual cart item:

#### 'cartItem'

`ccrz__E_CartItem__c` to evaluate.

#### ccrz.ccApiAttribute.ATTRIBUTES

`Map<String, Object>` that describes `ccrz__E_Attribute__c` records related to the queried products.

#### ccrz.ccApiProduct.PRODUCTPRICINGDATA

`Map<String, Object>` of pricing data for all queried products. This map can contain seller, subscription term, attribute, and standard pricing information for the queried products.

#### 'parentItem'

The parent `ccrz__E_CartItem__c` of 'cartItem'.

2. For each cart item, the method creates another intermediate map and calls the `ccrz.ccLogicCartPrice.checkCartItemValidity` inner method.
3. If the cart item is included in a *complex product*, the method adds the item and its parent to another intermediate map. A complex product can be a bundle, kit, dynamic kit, assembly product, or a cart item where `ccrz__PricingType__c` is `ATTRGROUP`.

### complexCartItems

`Map<String, List<ccrz__E_CartItem__c>>`, keyed by parent cart item ID, that describes major cart line items and its associated minor cart line items.

4. After evaluating all cart items, the method passes the `complexCartItems` map to the `ccrz.ccLogicCartPrice.evaluateComplexProducts` inner method.

## Outputs

This method returns the working `Map<String, Object>` after the appropriate inner methods complete, and sets the following keys:

### 'cartItems'

`Map<String, ccrz__E_CartItem__c>` of cart items passed in from 'cartItemsToPrice' and updated with the result of the inner methods.

 **Note:** If a cart item's parent is a bundle, the method sets the following fields on the item's `ccrz__E_CartItem__c` record:

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__Price__c</code>	0.00
<code>ccrz__SubAmount__c</code>	0.00

### 'cartToUpdate'

A new `ccrz__E_Cart__c` record where `ccrz__RepricedDate__c` is set.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[Salesforce Help Create a Bundle](#)

## ccrz.ccLogicCartPrice.priceltem

Begins the For loop within the `ccrz.ccLogicCartPrice.applyPricing` method by evaluating the current cart item and determining the appropriate pricing logic.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> priceItem(Map<String, Object>)
```

## Inputs (Required)

This method accepts the `pricingInputs` `Map<String, Object>` from the `ccrz.ccLogicCartPrice.applyPricing` method. The map can include the following keys:

### 'cartItem'

`ccrz__E_CartItem__c` to evaluate.

### `ccrz.ccApiProduct.PRODUCTPRICINGDATA`

`Map<String, Object>` of pricing data for all queried products. This map can contain seller, subscription term, attribute, and standard pricing information for the queried products.

### 'parentItem'

The parent `ccrz__E_CartItem__c` of 'cartItem'.

## Outputs

This method returns the input `Map<String, Object>`, updated with the results of the appropriate pricing logic:

Scenario	Result
<code>ccrz.ccApiProduct.PRODUCTPRICINGDATA</code> is empty.	The current user isn't entitled to the product. This method sets the cart item's price to <code>-1</code> , which causes the <code>ccrz.ccLogicCartPrice.checkCartItemValidity</code> method to set the cart item's <code>ccrz__ItemStatus__c</code> to <code>Invalid</code> .
<code>ccrz.ccApiProduct.PRODUCTPRICINGDATA</code> contains seller data	Set the cart item's price from the seller's price list item. If the seller isn't listed, this cart item becomes invalid.
The cart item's <code>ccrz__CartItemType__c</code> is <code>Major</code> and the cart item has a value for <code>ccrz__SubProdTerm__c</code> .	Call the <code>ccrz.ccLogicCartPrice.evaluateSubProdtermPricing</code> method to determine the cart item's price.
The cart item has a value for <code>ccrz__SelectedAttr__c</code> .	Evaluate whether <code>ccrz.ccApiProduct.PRODUCTPRICINGDATA</code> includes attribute pricing data. If so, call the <code>ccrz.ccLogicCartPrice.priceAttributeItem</code> method to determine the cart item's price.
The cart item's <code>ccrz__PricingType__c</code> is <code>ATTRGROUP</code> .	Set the following fields on the <code>ccrz__E_CartItem__c</code> record: <ul style="list-style-type: none"> <li><code>ccrz__OriginalItemPrice__c</code> is <code>0.00</code></li> <li><code>ccrz__Price__c</code> is <code>0.00</code></li> <li><code>ccrz__Quantity__c</code> is <code>1</code></li> <li><code>ccrz__SubAmount__c</code> is <code>0.00</code></li> </ul>
The cart item has a value for <code>ccrz__PricingModifierTypes__c</code> .	Evaluate the product's pricing tiers, and apply the correct price based on the item's quantity.
The cart item's <code>ccrz__CartItemType__c</code> is <code>Minor</code> and the cart item's parent has a subscription term.	Call the <code>ccrz.ccLogicCartPrice.applySubscriptionPercentages</code> method to update the cart item's price.

Scenario	Result
All other scenarios	Apply standard pricing and set the following fields on the <code>ccrz__E_CartItem__c</code> record: <ul style="list-style-type: none"> <li>• <code>ccrz__OriginalItemPrice__c</code></li> <li>• <code>ccrz__Price__c</code></li> <li>• <code>ccrz__SubAmount__c</code></li> </ul>

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccLogicCartPrice.evaluateSubProdtermPricing

Parses the subscription term for a cart item, and sets the item's corresponding price fields.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> evaluateSubProdtermPricing (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>` from the `ccrz.ccLogicCartPrice.priceItem` method. The map can include the following keys:

**'cartItem'**

`ccrz__E_CartItem__c` to evaluate.


**'spt4Price'**

`Map<String, Object>` that describes the `ccrz__E_SubProdTerm__c` record that the cart item references.



## Outputs

This method returns the input `Map<String, Object>` and updates the values of the following `ccrz__E_CartItem__c` fields.

 **Note:** By default, this method applies price information from the modifier subscription term with the lowest Sequence value.

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__OriginalItemPrice__c</code>	The same value as <code>ccrz__Price__c</code> .
<code>ccrz__Price__c</code>	The subscription's upfront (non-recurring) price.
<code>ccrz__RecurringPrice__c</code>	The subscription's recurring price.
<code>ccrz__RecurringPriceSubAmt__c</code>	The recurring price multiplied by quantity.
<code>ccrz__SubAmount__c</code>	The upfront price multiplied by quantity.

SEE ALSO:

[ccrz\\_\\_E\\_SubProdTerm\\_\\_c](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## `ccrz.ccLogicCartPrice.applySubscriptionPercentages`

Adjusts the price for a cart item that's a minor line item for a kit or dynamic kit that defines a subscription term. This method applies the parent subscription's upfront and recurring percentage discounts to the child cart item, which previously had only standard pricing logic applied.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> applySubscriptionPercentages (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>` from the `ccrz.ccLogicCartPrice.priceItem` method. The map can include the following keys:

### 'cartItem'

`ccrz__E_CartItem__c` that's a minor line item under a kit or dynamic kit.

### 'parent'

The parent `ccrz__E_CartItem__c` of 'cartItem'. This cart item must be a kit or dynamic kit that defines a subscription term.

### 'sptMap'

`Map<String, Object>` that describes the `ccrz__E_SubProdTerm__c` record that the cart item references.

## Outputs

This method returns the input `Map<String, Object>` and updates the values of the following `ccrz__E_CartItem__c` fields.



**Note:** By default, this method applies price information from the modifier subscription term with the lowest Sequence value.

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__Price__c</code>	The standard price of the cart item, adjusted by the subscription's upfront (non-recurring) percentage.
<code>ccrz__RecurringPrice__c</code>	The standard price of the cart item, adjusted by the subscription's recurring percentage.
<code>ccrz__RecurringPriceSubAmt__c</code>	The recurring price multiplied by quantity.
<code>ccrz__SubAmount__c</code>	The upfront price multiplied by quantity.

SEE ALSO:

[ccrz\\_\\_E\\_SubProdTerm\\_\\_c](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz.ccLogicCartPrice.priceAttributeItem

Parses JSON attribute pricing data for a cart item, and sets the item's corresponding price fields.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> priceAttributeItem(Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>` from the `ccrz.ccLogicCartPrice.priceItem` method. The map can include the following keys:

### 'cartItem'

`ccrz__E_CartItem__c` that's a minor line item under a kit or dynamic kit.

### ccrz.ccApiAttribute.ATTRIBUTES

`Map<String, ccrz__E_Attribute__c>` returned from the `ccrz.ccApiAttribute.fetch` query that the `ccrz.ccLogicCartPrice.fetchPricingData` method invoked.

### 'attributePricing'

`Map<String, Object>` that includes the deserialized JSON attribute pricing data from the corresponding price list item.

## Outputs

This method returns the input `Map<String, Object>` and updates the values of the following `ccrz__E_CartItem__c` fields.

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__Price__c</code>	The price specified by the cart item's applicable primary, secondary, or tertiary attribute.
<code>ccrz__SubAmount__c</code>	The price multiplied by quantity.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccApiAttribute.fetch](#)

[ccrz.ccLogicCartPrice.fetchPricingData](#)

## ccrz.ccLogicCartPrice.checkCartItemValidity

Continues the For loop within the `ccrz.ccLogicCartPrice.applyPricing` method by evaluating validation criteria and updating the cart item's status.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> checkCartItemValidity(Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>` from the `ccrz.ccLogicCartPrice.applyPricing` method. The map can include the following keys:

### 'cartItem'

`ccrz__E_CartItem__c` to evaluate.

### 'parentItem'

The parent `ccrz__E_CartItem__c` of 'cartItem'.


## Outputs

This method returns the input `Map<String, Object>` and sets the value of the `ccrz__E_CartItem__c` record's `ccrz__ItemStatus__c` field to `Invalid` for any of the following conditions:

- The cart item's price is negative.
- The corresponding product's Start Date or End Date is invalid for the current date.
- The product's `ccrz__ProductStatus__c` field has a value other than `Released`, and the product isn't a component of a bundle.

When the cart item is invalid, this method also sets the following fields:

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__OriginalItemPrice__c</code>	The same value as <code>ccrz__Price__c</code> .
<code>ccrz__Price__c</code>	-1
<code>ccrz__RecurringPrice__c</code>	null
<code>ccrz__RecurringPriceSubAmt__c</code>	null
<code>ccrz__SubAmount__c</code>	null

 **Note:** If the cart item has a parent cart item, this method also sets the parent item's status to `Invalid`.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartPrice.evaluateComplexProducts

Evaluates cart items that are included in complex products and sets the status of each cart item. A *complex product* can be a bundle, kit, dynamic kit, assembly product, or attribute-based product.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> evaluateComplexProducts (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>` from the `ccrz.ccLogicCartPrice.applyPricing` method. The map can include the following keys:

**'cartItems'**

`Map<String, ccrz__E_CartItem__c>` of cart items that have parent cart items, or cart items that are part of attribute-based products (where `ccrz__PricingType__c` is `ATTRGROUP`).

**'completeInputData'**

`Map<String, Object>` of the working map passed from `ccrz.ccLogicCartPrice.applyPricing`.

**'complexCartItems'**

`Map<String, List<ccrz__E_CartItem__c>>`, keyed by parent cart item ID, that describes major cart line items and its associated minor cart line items.

### Outputs

This method returns the input `Map<String, Object>` and sets the value of the `ccrz__E_CartItem__c` record's `ccrz__ItemStatus__c` field to `Invalid` for any of the following conditions:

- The cart item is a major line item for a complex product (other than an assembly product) but it doesn't have any child cart items.

- The cart item's `ccrz__PricingType__c` is `ATTRGROUP` and contains child cart items where the `ccrz__ItemStatus__c` is already set to `Invalid`. This method also sets the major line item to `Invalid`.
- The cart item is a major line item for a kit or dynamic kit and contains child cart items where the `ccrz__ItemStatus__c` is already set to `Invalid`. This method also sets the major line item to `Invalid`.

When the cart item is invalid, this method also sets the following fields:

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__OriginalItemPrice__c</code>	The same value as <code>ccrz__Price__c</code> .
<code>ccrz__Price__c</code>	-1
<code>ccrz__RecurringPrice__c</code>	null
<code>ccrz__RecurringPriceSubAmt__c</code>	null
<code>ccrz__SubAmount__c</code>	null

For a kit or dynamic kit, this method also evaluates any subscription terms applied to the major line item, and sets the following fields:

<code>ccrz__E_CartItem__c</code> Record	Field	Value
Major line item (composite product for a kit or dynamic kit)	<code>ccrz__RecurringPrice__c</code>	Calculated from the child cart item's <code>ccrz__RecurringPrice__c</code>
	<code>ccrz__RecurringPriceSubAmt__c</code>	Recurring price multiplied by quantity.
Child line item (component product for a kit or dynamic kit)	<code>ccrz__RecurringPrice__c</code>	null
	<code>ccrz__RecurringPriceSubAmt__c</code>	null

SEE ALSO:

[Salesforce Help Types of Products for B2B Commerce for Visualforce](#)

## `ccrz.ccLogicCartPrice.performExternalPricing`

Provides a placeholder for pricing a cart from an external source. By default, this method is empty and performs no default action.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> performExternalPricing(Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCartPrice.applyPricing` method completes.

This method executes only when both of the following conditions are satisfied:

- The Automatically Calculate Price Adjustment storefront setting (`AutoCalcPriceAdjust__c`) is enabled.
- The working map includes the `ccrz.ccApiCart.FETCH_EXTERNAL_PRICING` key with a value of `true`.

By default, this method doesn't evaluate any input data.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

## ccrz.ccLogicCartPrice.performDML

Removes cart items, combines externally priced cart items with other cart items, and finally executes the DML `upsert` operation for the updated cart item records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> performDML(Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the appropriate method completes:

- `ccrz.ccLogicCartPrice.performExternalPricing` when external pricing applies
- `ccrz.ccLogicCartPrice.applyPricing` otherwise

This method requires the following keys in the working map:

**'cartItemsToDelete'**

List<`ccrz__E_CartItem__c`> of the items to remove from the working map. By default, the items to delete are those records that meet any of the following criteria:

- The cart item's quantity changed to 0.
- The cart item's `ccrz__ProductType__c` field has the value Coupon.

**'cartItemsToPrice'**

Map<String, `ccrz__E_CartItem__c`> of the items that previous methods have priced and that are ready for upsert.

This method also calls the `ccrz.ccLogicCartPrice.applyCoupons` inner method for adding a coupon cart item after the `upsert` operation completes.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**'externalCartItems'**

List<`ccrz__E_CartItem__c`> of the items where `ccrz__PricingType__c` is external. The previous pricing methods in this process chain excluded these items, but this method adds these items back to 'cartItemsToPrice' for upsert.

## Outputs

This method returns the working Map<String, Object> after the `upsert` operation completes and after invoking the `ccrz.ccLogicCartPrice.applyCoupons` method.

SEE ALSO:

[Upserting Records](#)

## ccrz.ccLogicCartPrice.applyCoupons

Creates a cart item for a coupon, and applies the coupon's discount to the cart.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



- In API version 10 or later, this method calls the `ccrz.ccApiCoupon.apply` method, which creates the coupon cart item, applies the discount, and executes the corresponding DML operations.
- In API version 9 or earlier, this method calls the `ccrz.cc_util_Coupon.applyCouponsToCart` method, which creates the coupon cart item and applies the discount.

## Signature

```
global virtual Map<String, Object> applyCoupons (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartPrice.performDML` method.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### 'cartItemsToPrice'

`Map<String, ccrz__E_CartItem__c>` of items in the cart.

### 'externalCartItems'

`List<ccrz__E_CartItem__c>` of the items where `ccrz__PricingType__c` is `external`. This method combines 'externalCartItems' with 'cartItemsToPrice' so that the coupon evaluates all items in the cart.

## Outputs

This method returns the working `Map<String, Object>` after invoking the `ccrz.ccApiCoupon.apply` method.

SEE ALSO:

[ccrz.ccApiCoupon.apply](#)

[Salesforce Help Offer Coupons for Discounts on Storefront Purchases](#)

## ccrz.ccLogicCartPrice.refetch

Call the `ccrz.ccApiCart.fetch` method for fetching the repriced and updated cart and cart items.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)


This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartPrice.performDML` method.

This method requires the following keys in the working map:

### `ccrz.ccApi.SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to query.

### `ccrz.ccApiCart.CART_ID`

String that specifies the ID of the cart to query.

 **Note:** Include `ccrz.ccApiCart.CART_ENCID` or `ccrz.ccApiCart.CART_ID`, but not both.

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiCart.fetch` outputs.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## `ccrz.ccLogicCartPrice.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPrice.refetch` method.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`
- `'cartItemList'`
- `'cartItemsToDelete'`
- `'cartItemsToPrice'`
- `'cartItemsToUpdate'`
- `'cartToPrice'`
- `'externalCartItems'`
- `'majorsWExtChild'`
- `'productPricingData'`
- `'skpPreProc'`

## ccrz.ccLogicCartPriceAsync

Define the business logic for converting a cart to an order using asynchronous Apex jobs.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Global API Caller

`ccrz.ccApiCart.priceAsync`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.priceAsync` method call.

#### [retrieveCartData](#)

Calls the `ccrz.ccApiCart.fetch` method for querying the `ccrz__E_Cart__c` record to price. Finally, this method also queries the cart's related `ccrz__E_CartCoupon__c` records, but doesn't query the cart's items. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

#### [validateLastPriceDate](#)

Evaluates whether the `ccrz__RepricedDate__c` field on the `ccrz__E_Cart__c` record is considered current, depending on the value of the Optimize Cart Repricing storefront configuration. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

#### [prepareAsyncPrice](#)

Wraps inner methods that execute synchronously during the initial `ccrz.ccApiCart.priceAsync` call. These methods determine whether to price the cart. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.priceAsync` method call.

#### [enqueuePricing](#)

Calls the `System.enqueueJob` method for adding the queueable `ccLogicCartPriceEnqueue` class to the asynchronous Apex job queue. To call a custom queueable class, override this method with custom subscriber code.

#### [updateCartHeader](#)

Sets fields on the `ccrz__E_Cart__c` record to reference the asynchronous Apex job ID, update the cart's status, and set the repriced date. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

#### [performDMLOnCart](#)

Performs the DML `update` operation on the `ccrz__E_Cart__c` record to price. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

#### [executeAsyncPrice](#)

Wraps inner methods that execute during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` calls. These methods process a subset of `ccrz__E_CartItem__c` records from the source `ccrz__E_Cart__c`.

#### [retrieveCartItems](#)

Determines how many `ccrz__E_CartItem__c` records to price during the recursive job, and then calls the `ccrz.ccApiCart.fetchCartItems` method for querying them. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

#### [prepareCartItems](#)

Identifies `ccrz__E_CartItem__c` records to pass to or exclude from the subsequent pricing methods. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [fetchPricingData](#)

Calls the `ccrz.ccApiProduct.fetch` method and returns product pricing data for the cart items to reprice. If the fetched product data includes attributes, this method also calls the `ccrz.ccApiAttribute.fetch` method. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [applyPricing](#)

Loops through the cart items to reprice, and calls the appropriate inner method for setting each cart item's price. This method also ensures that major and minor line items remain associated with each other. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [priceltem](#)

Begins the For loop within the `ccrz.ccLogicCartPriceAsync.applyPricing` method by evaluating the current cart item and determining the appropriate pricing logic. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [evaluateSubProdtermPricing](#)

Parses the subscription term for a cart item, and sets the item's corresponding price fields. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [applySubscriptionPercentages](#)

Adjusts the price for a cart item that's a minor line item for a kit or dynamic kit that defines a subscription term. This method applies the parent subscription's upfront and recurring percentage discounts to the child cart item, which previously had only standard pricing logic applied. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [priceAttributeltem](#)

Parses JSON attribute pricing data for a cart item, and sets the item's corresponding price fields. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [checkCartItemValidity](#)

Continues the For loop within the `ccrz.ccLogicCartPriceAsync.applyPricing` method by evaluating validation criteria and updating the cart item's status. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [evaluateComplexProducts](#)

Evaluates cart items that are included in complex products and sets the status of each cart item. A *complex product* can be a bundle, kit, dynamic kit, assembly product, or attribute-based product. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [performExternalPricing](#)

Provides a placeholder for pricing a cart from an external source. By default, this method is empty and performs no default action. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [performDMLOnItems](#)

Removes cart items, combines externally priced cart items with other cart items, and finally executes the DML `upsert` operation for the updated `ccrz__E_CartItem__c` records. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### [enqueueNext](#)

Determines whether more major line item `ccrz__E_CartItem__c` records remain to be priced, and calls `ccrz.ccLogicCartPriceAsync.enqueuePricing` for queueing the next job. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### applyCoupons

Creates a cart item for a coupon, and applies the coupon's discount to the cart. This method calls the `ccrz.ccApiCoupon.apply` method, which creates the coupon cart item, applies the discount, and executes the corresponding DML operations. This method executes only during the final recursive `ccrz.ccApiCart.priceAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

### enqueuePlace

Calls the `System.enqueueJob` method for adding the queueable `ccLogicCartPriceEnqueuePlace` class to the asynchronous Apex job queue. To call a custom queueable class, override this method with custom subscriber code. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls, and only when pricing occurs as part of the `ccrz.ccApiCart.placeAsync` logic chain.

### updateOnError

Updates the fields on the `ccrz__E_Cart__c` record when an exception occurs during asynchronous cart pricing. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

### handleCartPriceException

Catches a `ccrz.ccLogicCartPriceAsync.CartPriceAsyncException` that occurs during the logic chain, and logs the exception message. If the pricing logic is executing as part of a `ccrz.ccApiCart.placeAsync` call, this method also catches a `ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException` to ensure that the error propagates and ends order placement.

## ccrz.ccLogicCartPriceAsync.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.priceAsync` method.

### Logic Performed

Asynchronous cart pricing consists of calls to the `ccrz.ccApiCart.priceAsync` method that occur in two phases:

### Initial synchronous call


The initial `ccrz.ccApiCart.priceAsync` call enters a *synchronous* portion, where this method calls into the following inner methods:

1. `processInput`
2. `retrieveCartData`
3. `validateLastPriceData`
4. `prepareAsyncPrice`, which calls other inner methods

### Recursive, asynchronous calls

Each recursive `ccrz.ccApiCart.priceAsync` call enters an *asynchronous* portion, where this method calls into `executeAsyncPrice`, which calls other inner methods.

The `prepReturn` method executes after all recursive jobs have completed.

 **Note:** The working map maintains an input key, `ccrz.ccApiCart.PRICE_ASYNC_FINISH`, that indicates the status of the logic chain. When this key is `false`, the logic is executing the synchronous portion. When this key becomes `true`, the logic is executing the asynchronous portion.

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

This method rethrows any `ccrz.ccLogicCartPriceAsync.CartPriceAsyncException` instances that occur during the logic chain, which ends processing and rolls back transactions. The `ccrz.ccLogicCartPriceAsync.handleCartPriceException` method catches these exceptions.

## ccrz.ccLogicCartPriceAsync.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.priceAsync` method call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPriceAsync.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to price.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to price.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean that's assigned an intermediate value of `false`. Subsequent methods in the logic chain update this value to `true` when the logic completes successfully.

**ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartPrice.prepReturn method` removes these intermediate keys from the working map.

## ccrz.ccLogicCartPriceAsync.retrieveCartData

Calls the `ccrz.ccApiCart.fetch` method for querying the `ccrz__E_Cart__c` record to price. Finally, this method also queries the cart's related `ccrz__E_CartCoupon__c` records, but doesn't query the cart's items. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> retrieveCartData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the appropriate method:

**Initial synchronous call**

After `ccrz.ccLogicCartPriceAsync.processInput` completes

**Recursive, asynchronous calls**

From `ccrz.ccLogicCartPriceAsync.executeAsyncPrice`

This method requires either of the following keys in the working map:


**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to price.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to price.



 **Note:** If neither key is specified, this method returns a `ccrz.ccLogicCartPriceAsync.CartPriceAsyncException`.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### `ccrz.ccApiCart.CONTINUE_ASYNC_PLACE_AFTER_PRICING`

Boolean

Value	Usage
<code>true</code>	This method is executing as part of the <code>ccrz.ccApiCart.placeAsync</code> method. When this method completes, it calls <code>ccrz.ccApiCart.placeAsync</code> recursively to continue the order placement process.
<code>false</code> (default)	This method is executing outside of the <code>ccrz.ccApiCart.placeAsync</code> method.

### `ccrz.ccApiCart.PRICE_ASYNC_FINISH`

Boolean

Value	Usage
<code>true</code>	The current method execution is a recursive, asynchronous call that continues the cart pricing after the initial call.
<code>false</code> (default)	The current method execution is the synchronous portion of the initial call that begins the cart pricing process.

## Logic Performed

This method completes the following actions:

1. Evaluates whether the input data includes the following inputs, and sets an appropriate value for the `ccrz.ccApiCart.fetch` method's `ccrz.ccApiCart.CARTSTATUS` input:
  - If the input data specifies `ccrz.ccApiCart.PRICE_ASYNC_FINISH => true`, the `ccrz.ccApiCart.fetch` method queries for a cart where the `ccrz__CartStatus__c` value is Repricing.
  - If the input data specifies `ccrz.ccApiCart.CONTINUE_ASYNC_PLACE_AFTER_PRICING => true`, the `ccrz.ccApiCart.fetch` method queries for a cart where the `ccrz__CartStatus__c` value is Processing.
  - Otherwise, the `ccrz.ccApiCart.fetch` method queries for a cart where the `ccrz__CartStatus__c` value is Open.
2. Calls the `ccrz.ccApiCart.fetch` method for returning the cart to price.

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiCart.fetch` outputs, and sets the following intermediate keys:

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the fetched cart.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the fetched cart.

**ccrz.ccApiCart.HAS\_COUPON\_CODE**

Boolean

Value	Usage
true	The cart has a coupon applied.
false	The cart doesn't have any coupon applied.

**ccrz.ccLogicCartPriceAsync.CART\_TO\_PRICE**

[ccrz\\_\\_E\\_Cart\\_\\_c](#) record that represents the cart to price.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz.ccApiCart.fetch](#)

**ccrz.ccLogicCartPriceAsync.validateLastPriceDate**

Evaluates whether the `ccrz__RepricedDate__c` field on the `ccrz__E_Cart__c` record is considered current, depending on the value of the Optimize Cart Repricing storefront configuration. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

**Signature**

```
global virtual Map<String, Object> validateLastPriceDate (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.retrieveCartData` method completes.

This method requires the following key in the working map:

**ccrz.ccApiCart.PRICE\_ASYNC\_FINISH**

Boolean

Value	Usage
<i>true</i>	The current method execution is a recursive, asynchronous call that continues the cart pricing after the initial call.
<i>false</i> (default)	The current method execution is the synchronous portion of the initial call that begins the cart pricing process.

**ccrz.ccLogicCartPriceAsync.CART\_TO\_PRICE**

`ccrz__E_Cart__c` record that represents the cart to price.

**Inputs (Optional)**

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiCart.HAS\_COUPON\_CODE**

Boolean

Value	Usage
<i>true</i>	The cart has a coupon applied.
<i>false</i>	The cart doesn't have any coupon applied.

**ccrz.ccApiCart.SKIP\_CART\_PRICING**

Boolean

Value	Usage
<i>true</i>	Skip the logic in this method, and immediately return the working <code>Map&lt;String, Object&gt;</code> .
<i>false</i> (default)	Execute the logic in this method.

**Logic Performed**

This method completes the following actions:

1. Evaluates whether input data already specifies a value for `ccrz.ccApiCart.SKIP_CART_PRICING`. Otherwise, the method continues.
2. Finds the value of the Cart storefront configuration module's Optimize Cart Repricing setting (`c.optCartReprice`).
3. Evaluates whether any of the following criteria are satisfied, and if so, sets a value for `ccrz.ccApiCart.SKIP_CART_PRICING`:
  - If the `ccrz.ccApiCart.HAS_COUPON_CODE` input is *true*, the method sets `ccrz.ccApiCart.SKIP_CART_PRICING` to *false*.
  - If the cart's `ccrz__RepricedDate__c` date matches the current date, the method sets `ccrz.ccApiCart.SKIP_CART_PRICING` to *true*.
  - If the cart's `ccrz__RepricedDate__c` is more recent than the Valid Prices Date storefront setting, the method sets `ccrz.ccApiCart.SKIP_CART_PRICING` to *true*.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCart.SKIP_CART_PRICING`

Boolean

Value	Usage
<code>true</code>	Skip the inner methods of the <code>ccrz.ccLogicCartPriceAsync.prepareAsyncPrice</code> method.
<code>false</code>	Execute the next method in the process chain.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[Salesforce Help Optimize How Frequently the Shopping Cart Page Calls Price Logic](#)

## `ccrz.ccLogicCartPriceAsync.prepareAsyncPrice`

Wraps inner methods that execute synchronously during the initial `ccrz.ccApiCart.priceAsync` call. These methods determine whether to price the cart. This method executes only during the synchronous portion of the initial `ccrz.ccApiCart.priceAsync` method call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> prepareAsyncPrice (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.validateLastPriceDate` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiCart.PRICE_ASYNC_FINISH`

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	The current method execution is a recursive, asynchronous call that continues the cart pricing after the initial call.

Value	Usage
<code>false</code> (default)	The current method execution is the synchronous portion of the initial call that begins the cart pricing process.

**ccrz.ccApiCart.SKIP\_CART\_PRICING**


Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Skip the inner methods of the <code>ccrz.ccLogicCartPriceAsync.prepareAsyncPrice</code> method.
<code>false</code>	Execute the next method in the process chain.

## Logic Performed

During this synchronous call, this method invokes other methods of the `ccrz.ccLogicCartPriceAsync` class in the following order:

1. `enqueuePricing`
2. `updateCartHeader`
3. `performDMLOnCart`

 **Note:** If `ccrz.ccApiCart.SKIP_CART_PRICING` is `true`, all of these methods are skipped.

## Outputs

This method returns the working `Map<String, Object>` after all inner methods have completed. Refer to the individual method reference for more information about each method's outputs.

SEE ALSO:

[ccrz.ccApiCart.priceAsync](#)

## ccrz.ccLogicCartPriceAsync.enqueuePricing

Calls the `System.enqueueJob` method for adding the queueable `ccLogicCartPriceEnqueue` class to the asynchronous Apex job queue. To call a custom queueable class, override this method with custom subscriber code.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> enqueuePlace (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the appropriate method:

### Initial synchronous call

From `ccrz.ccLogicCartPriceAsync.prepareAsyncPrice`

### Recursive, asynchronous calls

From `ccrz.ccLogicCartPriceAsync.enqueueNext`

This method requires the following key in the working map:

### `ccrz.ccApiCart.SKIP_CART_PRICING`

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Skip the inner methods of the <code>ccrz.ccLogicCartPriceAsync.prepareAsyncPrice</code> method.
<code>false</code>	Execute the next method in the process chain.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCart.PRICE_ASYNC_JOB_ID`

ID of the `AsyncApexJob` that represents the asynchronous Apex job that the `System.enqueueJob` method returned.

SEE ALSO:

[enqueueJob \(queueableObj\)](#)

[Queueable Apex](#)

## `ccrz.ccLogicCartPriceAsync.updateCartHeader`

Sets fields on the `ccrz__E_Cart__c` record to reference the asynchronous Apex job ID, update the cart's status, and set the repriced date. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> updateCartHeader (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the appropriate method completes:

### Initial synchronous call

After `ccrz.ccLogicCartPriceAsync.enqueuePricing`

### Recursive, asynchronous calls

After `ccrz.ccLogicCartPriceAsync.enqueueNext`

This method requires the following key in the working map:

**ccrz.ccLogicCartPriceAsync.CART\_TO\_PRICE**  
`ccrz__E_Cart__c` record that represents the cart to price.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiCart.CONTINUE\_ASYNC\_PLACE\_AFTER\_PRICING**  
 Boolean

Value	Usage
<i>true</i>	This method is executing as part of the <code>ccrz.ccApiCart.placeAsync</code> method. When this method completes, it calls <code>ccrz.ccApiCart.placeAsync</code> recursively to continue the order placement process.
<i>false</i> (default)	This method is executing outside of the <code>ccrz.ccApiCart.placeAsync</code> method.

**ccrz.ccApiCart.NEXT\_PRICING\_JOB\_ENQUEUED**  
 Boolean

Value	Usage
<i>true</i>	Another asynchronous Apex job is enqueued for continuing the pricing process.
<i>false</i>	The current asynchronous Apex Job is the final job for the pricing process.

**ccrz.ccApiCart.PRICE\_ASYNC\_FINISH**  
 Boolean

Value	Usage
<i>true</i>	The current method execution is a recursive, asynchronous call that continues the cart pricing after the initial call.
<i>false</i> (default)	The current method execution is the synchronous portion of the initial call that begins the cart pricing process.

**ccrz.ccApiCart.PRICE\_ASYNC\_JOB\_ID**

ID of the `AsyncApexJob` that represents the asynchronous Apex job that the `System.enqueueJob` method returned.

**Logic Performed**

This method evaluates `ccrz.ccApiCart.PRICE_ASYNC_FINISH`, and performs different actions depending on whether the current method execution is the initial or a recursive call:

**Initial synchronous call**

1. Sets up a new `ccrz__E_Cart__c` record with the following values:

Field	Value
<code>ccrz__AsynchronousJobID__c</code>	The ID that <code>ccrz.ccApiCart.PRICE_ASYNC_JOB_ID</code> specifies
<code>ccrz__CartStatus__c</code>	<i>Repricing</i>
<code>ccrz__RepricedDate__c</code>	The current date and time

2. Evaluates whether the `ccrz.ccApiCart.CONTINUE_ASYNC_PLACE_AFTER_PRICING` input is `true`. If so, this method adjusts `ccrz__CartStatus__c` to *Processing*.

**Recursive, asynchronous calls**

1. Evaluates whether the `ccrz.ccApiCart.NEXT_PRICING_JOB_ENQUEUED` input is `true`:
  - If `true`, the method sets up a new `ccrz__E_Cart__c` record with the following values:

Field	Value
<code>ccrz__AsynchronousJobID__c</code>	The ID that <code>ccrz.ccApiCart.PRICE_ASYNC_JOB_ID</code> specifies

- Otherwise, the method sets up a new `ccrz__E_Cart__c` record with the following values:

Field	Value
<code>ccrz__AsynchronousJobID__c</code>	The ID that <code>ccrz.ccApiCart.PRICE_ASYNC_JOB_ID</code> specifies
<code>ccrz__CartStatus__c</code>	<i>Open</i>

- Evaluates whether the `ccrz.ccApiCart.CONTINUE_ASYNC_PLACE_AFTER_PRICING` input is `true`. If so, this method adjusts `ccrz__CartStatus__c` to *Processing*.


**Outputs**

This method returns the working `Map<String, Object>`, and sets the following key:



**ccrz.ccLogicCartPriceAsync.CART\_TO\_UPDATE**

`ccrz__E_Cart__c` record that represents the new cart record, with the same ID as the source `ccrz__E_Cart__c` record that the `ccrz.ccLogicCartPriceAsync.CART_TO_PRICE` input specified. This new cart record has the appropriate fields set, and is ready to insert.

 **Note:** Creating a separate `ccrz__E_Cart__c` record avoids accidental changes to the source `ccrz__E_Cart__c` record that the `ccrz.ccLogicCartPriceAsync.CART_TO_PRICE` input specified.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

## ccrz.ccLogicCartPriceAsync.performDMLOnCart

Performs the DML `update` operation on the `ccrz__E_Cart__c` record to price. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> performDMLOnOrder (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.updateCartHeader` method completes.

This method requires the following keys in the working map:

**ccrz.ccLogicCartPriceAsync.CART\_TO\_UPDATE**

`ccrz__E_Cart__c` record that represents the new cart record that the `ccrz.ccLogicCartPriceAsync.updateCartHeader` method created.

### Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccApiCart.PRICE\_ASYNC\_FINISH**

Boolean

Value	Usage
<code>true</code>	The current method execution is a recursive, asynchronous call that continues the cart pricing after the initial call.

Value	Usage
<i>false</i> (default)	The current method execution is the synchronous portion of the initial call that begins the cart pricing process.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartPriceAsync.executeAsyncPrice

Wraps inner methods that execute during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` calls. These methods process a subset of `ccrz__E_CartItem__c` records from the source `ccrz__E_Cart__c`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> executeAsyncPrice (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.processInput` method completes.

This method requires the following key in the working map:

**ccrz.ccApiCart.PLACE\_ASYNC\_FINISH**


Boolean that must be `true` for this method to execute.

Value	Usage
<i>true</i>	The current method execution is a recursive, asynchronous call that continues the cart pricing after the initial call.
<i>false</i>	The current method execution is the synchronous portion of the initial call that begins the cart pricing process.


## Logic Performed

During each recursive, asynchronous call, this method invokes other methods of the `ccrz.ccLogicCartPriceAsync` class in the following order for pricing a subset of `ccrz__E_CartItem__c` records:

1. `retrieveCartData`
2. `validateLastPriceDate`
3. `retrieveCartItems`
4. `prepareCartItems`
5. `fetchPricingData`
6. `applyPricing`, which calls the following inner methods:
  - a. `priceItem` for each cart item, which calls into the appropriate method for pricing different product scenarios:
    - `evaluateSubProdTermPricing`
    - `applySubscriptionPercentages`
    - `priceAttributeItem`
  - b. `checkCartItemValidity` for each cart item
  - c. `evaluateComplexProducts`
7. `performExternalPricing`

 **Note:** The logic chain specifies a database savepoint after the `performExternalPricing` method completes to allow for rollback when errors occur. This rollback applies only to the batch of items that the current job is processing, and not *all* items.

8. `performDMLOnItems`
9. `enqueueNext`
10. `updateCartHeader`
11. `enqueuePlace`

 **Note:** If any exceptions occur during asynchronous processing, the `updateOnError` method executes.

## Outputs

This method returns the working `Map<String, Object>` after all inner methods have completed. Refer to the individual method reference for more information about each method's outputs.

SEE ALSO:

[ccrz.ccApiCart.priceAsync](#)

## `ccrz.ccLogicCartPriceAsync.retrieveCartItems`

Determines how many `ccrz__E_CartItem__c` records to price during the recursive job, and then calls the `ccrz.ccApiCart.fetchCartItems` method for querying them. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> retrieveCartItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.validateLastPriceDate` method completes.

This method requires the following key in the working map:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to price.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiCart.CARTITEMLASTID**

String that specifies the Salesforce ID of the final major cart item that was processed during the previous recursive job.

**ccrz.ccApiCart.CART\_PRICE\_BATCH\_SIZE**

Integer that specifies the number of major cart items to price during each asynchronous Apex job.




### Note:

- If you don't specify a value for this input, the method defaults to 200 major cart items.
- The job prices any minor cart items that are associated with the major cart items. As a result, the total number of priced `ccrz__E_CartItem__c` records is higher when the cart contains complex products. When a pricing scenario includes a large number of complex products, consider specifying a lower batch size to help fine-tune performance.


## Logic Performed

This method completes the following actions:

1. Evaluates whether the input data already specifies a value for `ccrz.ccApiCart.CART_PRICE_BATCH_SIZE`. Otherwise, the method sets a default value of 200 `ccrz__E_CartItem__c` records to evaluate during the recursive job.
2. Evaluates whether the input data specifies `ccrz.ccApiCart.CARTITEMLASTID`:
  - If so, the method starts processing `ccrz__E_CartItem__c` records from the final ID evaluated during the previous job.
  - Otherwise, the method starts processing `ccrz__E_CartItem__c` records from the beginning of the `ccrz__E_Cart__c` record's list of items.
3. Executes a direct query of major item `ccrz__E_CartItem__c` records for the source `ccrz__E_Cart__c`.

 **Note:** To keep the order of `ccrz__E_CartItem__c` records consistent for subsequent recursive jobs, the query uses a `SOQL ORDER BY Id` clause.

4. Calls the `ccrz.ccApiCart.fetchCartItems` method for returning the batch of cart items for the current job.

 **Note:** The method also queries any minor cart items that are associated with the major cart items. As a result, the total number of priced `ccrz__E_CartItem__c` records is higher than the specified batch size when the cart contains complex products.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCart.CARTITEMLASTID`

String that specifies the Salesforce ID of the final major cart item that was processed during the current recursive job.

### `ccrz.ccLogicCartPriceAsync.CART_ITEM_LIST`

`List<ccrz__E_CartItem__c>` that contains the major and minor line items that the `ccrz.ccApiCart.fetchCartItems` returned.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccApiCart.fetchCartItems](#)

## `ccrz.ccLogicCartPriceAsync.prepareCartItems`

Identifies `ccrz__E_CartItem__c` records to pass to or exclude from the subsequent pricing methods. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> prepareCartItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.retrieveCartItems` method completes.

This method requires the following keys in the working map:

**ccrz.ccLogicCartPriceAsync.CART\_ITEM\_LIST**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> that contains the major and minor line items to evaluate.

**ccrz.ccLogicCartPriceAsync.CART\_TO\_PRICE**

[ccrz\\_\\_E\\_Cart\\_\\_c](#) record that represents the cart to price.

## Outputs

This method returns the working Map<String, Object>, and sets the following keys:

**'attributes'**

Set<String> of attribute IDs, populated from each cart item's [ccrz\\_\\_PrimaryAttr\\_\\_c](#), [ccrz\\_\\_SecondaryAttr\\_\\_c](#), and [ccrz\\_\\_TertiaryAttr\\_\\_c](#) fields.

**ccrz.ccLogicCartPriceAsync.EXTERNALLY\_PRICED\_ITEMS**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> of the items where [ccrz\\_\\_PricingType\\_\\_c](#) is external. These items are excluded from the subsequent pricing methods.


**ccrz.ccLogicCartPriceAsync.ITEMS\_TO\_DELETE**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> of the items to remove from the working map. By default, this method adds a cart item to this list when the item meets any of the following criteria:

- The cart item's quantity changed to 0.
- The cart item's [ccrz\\_\\_ProductType\\_\\_c](#) field has the value Coupon.

**ccrz.ccLogicCartPriceAsync.ITEMS\_TO\_PRICE**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> of the items to pass to subsequent pricing methods. For each cart item, this method updates the values of the following fields:

<a href="#">ccrz__E_CartItem__c</a> Field	Value
<a href="#">ccrz__ItemStatus__c</a>	Available
<a href="#">ccrz__Price__c</a>	-1
	 <b>Note:</b> This value is an initial value. The inner methods of <a href="#">ccrz.ccLogicCartPriceAsync.applyPricing</a> evaluate each cart item and update this value.

**ccrz.ccLogicCartPriceAsync.MAJORS\_WITH\_EXTERNAL\_CHILDREN**

List<String> of cart item IDs that are major line items with child cart items where [ccrz\\_\\_PricingType\\_\\_c](#) is external. These items are excluded from the subsequent pricing methods.

**ccrz.ccLogicCartPriceAsync.PRICING\_INPUT\_MAP**

Map<String, Object> of input data for a [ccrz.ccApiProduct.fetch](#) query, which the [ccrz.ccLogicCartPriceAsync.fetchPricingData](#) method executes. This map includes:

- Available data from other return data keys—such as 'products', 'attributes', 'sellers', and 'spts'—which specify parameters for the fetch query.
- The key value [ccrz.ccApiProduct.PARAM\\_INCLUDE\\_PRICING](#) => `true`, which ensures that the fetch query returns [ccrz.ccApiProduct.PRODUCTPRICINGDATA](#) and [ccrz.ccApiAttribute.IDS](#), if applicable.

**'products'**

Set<String> of product IDs, populated from each cart item's [ccrz\\_\\_Product\\_\\_c](#) field.

**'sellers'**

Set<String> of seller IDs, populated from each cart item's `ccrz__Seller__c` field.

**'spts'**

Set<String> of subscription term IDs, populated from each cart item's `ccrz__SubProdTerm__c` field.

## SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccApiProduct.fetch](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz.ccLogicCartPriceAsync.fetchPricingData

Calls the `ccrz.ccApiProduct.fetch` method and returns product pricing data for the cart items to reprice. If the fetched product data includes attributes, this method also calls the `ccrz.ccApiAttribute.fetch` method. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> fetchPricingData (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.prepareCartItems` method completes.

This method requires the following key in the working map:

**ccrz.ccLogicCartPriceAsync.PRICING\_INPUT\_MAP**

Map<String, Object> of input data for a `ccrz.ccApiProduct.fetch` query. This map includes:

- Products, attributes, sellers, and subscription terms parameters, if applicable.
- The key value `ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true`, which ensures that the fetch query returns `ccrz.ccApiProduct.PRODUCTPRICINGDATA` and `ccrz.ccApiAttribute.IDS`, if applicable.

### Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiProduct.fetch` outputs. In particular, the subsequent pricing methods evaluate the following output keys:

**ccrz.ccApiAttribute.ATTRIBUTES**

Map<String, Object> that describes `ccrz__E_Attribute__c` records related to the queried products. The `ccrz.ccApiAttribute.fetch` method returns this map.

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**

Map<String, Object> of pricing data for all queried products. This map can contain seller, subscription term, attribute, and standard pricing information for the queried products.

SEE ALSO:

[ccrz.ccApiProduct.fetch](#)

[ccrz.ccApiAttribute.fetch](#)

## ccrz.ccLogicCartPriceAsync.applyPricing

Loops through the cart items to reprice, and calls the appropriate inner method for setting each cart item's price. This method also ensures that major and minor line items remain associated with each other. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> applyPricing (Map<String, Object>)
```

### Inputs (Required)

This method accepts the Map<String, Object> after the `ccrz.ccLogicCartPriceAsync.fetchPricingData` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiAttribute.ATTRIBUTES**

Map<String, Object> that describes `ccrz__E_Attribute__c` records related to the queried products. The `ccrz.ccApiAttribute.fetch` method returns this map.

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**

Map<String, Object> of pricing data for all queried products. This map can contain seller, subscription term, attribute, and standard pricing information for the queried products.

**ccrz.ccLogicCartPriceAsync.ITEMS\_TO\_PRICE**

Map<String, `ccrz__E_CartItem__c`> of the items to price.

### Logic Performed

This method completes the following actions:



1. For each cart item, the method creates an intermediate map and passes it to the `ccrz.ccLogicCartPriceAsync.priceItem` inner method:

**pricingInputs**

Map<String, Object> that contains the following keys that the `ccrz.ccLogicCartPriceAsync.priceItem` inner method evaluates for an individual cart item:

**ccrz.ccApiAttribute.ATTRIBUTES**

Map<String, Object> that describes `ccrz__E_Attribute__c` records related to the queried products.

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**

Map<String, Object> of pricing data for all queried products. This map can contain seller, subscription term, attribute, and standard pricing information for the queried products.

**ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_CART\_ITEM**

`ccrz__E_CartItem__c` to evaluate.

**ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_PARENT\_ITEM**

The parent `ccrz__E_CartItem__c` of `ccrz.ccLogicCartPriceAsync.PRICE_ITEM_CART_ITEM`.

2. For each cart item, the method creates another intermediate map and calls the `ccrz.ccLogicCartPriceAsync.checkCartItemValidity` inner method.
3. If the cart item is included in a *complex product*, the method adds the item and its parent to another intermediate map. A complex product can be a bundle, kit, dynamic kit, assembly product, or a cart item where `ccrz__PricingType__c` is ATTRGROUP.

**complexCartItems**

Map<String, List<`ccrz__E_CartItem__c`>>, keyed by parent cart item ID, that describes major cart line items and its associated minor cart line items.

4. After evaluating all cart items, the method passes the `complexCartItems` map to the `ccrz.ccLogicCartPriceAsync.evaluateComplexProducts` inner method.

## Outputs

This method returns the working Map<String, Object> after the appropriate inner methods complete, and sets the following keys:

**ccrz.ccLogicCartPriceAsync.ITEMS\_TO\_PRICE**

Map<String, `ccrz__E_CartItem__c`> of the items passed in and updated with the result of the inner methods.

 **Note:** If a cart item's parent is a bundle, the method sets the following fields on the item's `ccrz__E_CartItem__c` record:

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__Price__c</code>	0.00
<code>ccrz__SubAmount__c</code>	0.00

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[Salesforce Help Create a Bundle](#)

## ccrz.ccLogicCartPriceAsync.priceltem

Begins the For loop within the `ccrz.ccLogicCartPriceAsync.applyPricing` method by evaluating the current cart item and determining the appropriate pricing logic. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> priceItem(Map<String, Object>)
```

### Inputs (Required)

This method accepts the `pricingInputs Map<String, Object>` from the `ccrz.ccLogicCartPriceAsync.applyPricing` method. The map can include the following keys:

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**

`Map<String, Object>` of pricing data for all queried products. This map can contain seller, subscription term, attribute, and standard pricing information for the queried products.

**ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_CART\_ITEM**

`ccrz__E_CartItem__c` to evaluate.

**ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_PARENT\_ITEM**

The parent `ccrz__E_CartItem__c` of `ccrz.ccLogicCartPriceAsync.PRICE_ITEM_CART_ITEM`.

### Outputs

This method returns the input `Map<String, Object>`, updated with the results of the appropriate pricing logic:

Scenario	Result
<code>ccrz.ccApiProduct.PRODUCTPRICINGDATA</code> is empty.	The current user isn't entitled to the product. This method sets the cart item's price to <code>-1</code> , which causes the <code>ccrz.ccLogicCartPriceAsync.checkCartItemValidity</code> method to set the cart item's <code>ccrz__ItemStatus__c</code> to <code>Invalid</code> .
<code>ccrz.ccApiProduct.PRODUCTPRICINGDATA</code> contains seller data	Set the cart item's price from the seller's price list item. If the seller isn't listed, this cart item becomes invalid.
The cart item's <code>ccrz__CartItemType__c</code> is <code>Major</code> and the cart item has a value for <code>ccrz__SubProdTerm__c</code> .	Call the <code>ccrz.ccLogicCartPriceAsync.evaluateSubProdtemPricing</code> method to determine the cart item's price.

Scenario	Result
The cart item has a value for <code>ccrz__SelectedAttr__c</code> .	Evaluate whether <code>ccrz.ccApiProduct.PRODUCTPRICINGDATA</code> includes attribute pricing data. If so, call the <a href="#">ccrz.ccLogicCartPriceAsync.priceAttributeItem</a> method to determine the cart item's price.
The cart item's <code>ccrz__PricingType__c</code> is <code>ATTRGROUP</code> .	Set the following fields on the <code>ccrz__E_CartItem__c</code> record: <ul style="list-style-type: none"> <li>• <code>ccrz__OriginalItemPrice__c</code> is 0.00</li> <li>• <code>ccrz__Price__c</code> is 0.00</li> <li>• <code>ccrz__Quantity__c</code> is 1</li> <li>• <code>ccrz__SubAmount__c</code> is 0.00</li> </ul>
The cart item has a value for <code>ccrz__PricingModifierTypes__c</code> .	Evaluate the product's pricing tiers, and apply the correct price based on the item's quantity.
The cart item's <code>ccrz__CartItemType__c</code> is <code>Minor</code> and the cart item's parent has a subscription term.	Call the <a href="#">ccrz.ccLogicCartPriceAsync.applySubscriptionPercentages</a> method to update the cart item's price.
All other scenarios	Apply standard pricing and set the following fields on the <code>ccrz__E_CartItem__c</code> record: <ul style="list-style-type: none"> <li>• <code>ccrz__OriginalItemPrice__c</code></li> <li>• <code>ccrz__Price__c</code></li> <li>• <code>ccrz__SubAmount__c</code></li> </ul>

## SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccLogicCartPriceAsync.evaluateSubProdtermPricing

Parses the subscription term for a cart item, and sets the item's corresponding price fields. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> evaluateSubProdtermPricing(Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>` from the `ccrz.ccLogicCartPriceAsync.priceItem` method. The map can include the following keys:

**ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_CART\_ITEM**


`ccrz__E_CartItem__c` to evaluate.

**ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_SUBPRODTERM**

`Map<String, Object>` that describes the `ccrz__E_SubProdTerm__c` record that the cart item references.

## Outputs

This method returns the input `Map<String, Object>` and updates the values of the following `ccrz__E_CartItem__c` fields.

 **Note:** By default, this method applies price information from the modifier subscription term with the lowest Sequence value.

ccrz__E_CartItem__c Field	Value
<code>ccrz__OriginalItemPrice__c</code>	The same value as <code>ccrz__Price__c</code> .
<code>ccrz__Price__c</code>	The subscription's upfront (non-recurring) price.
<code>ccrz__RecurringPrice__c</code>	The subscription's recurring price.
<code>ccrz__RecurringPriceSubAmt__c</code>	The recurring price multiplied by quantity.
<code>ccrz__SubAmount__c</code>	The upfront price multiplied by quantity.

SEE ALSO:

[ccrz\\_\\_E\\_SubProdTerm\\_\\_c](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz.ccLogicCartPriceAsync.applySubscriptionPercentages

Adjusts the price for a cart item that's a minor line item for a kit or dynamic kit that defines a subscription term. This method applies the parent subscription's upfront and recurring percentage discounts to the child cart item, which previously had only standard pricing logic applied. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> applySubscriptionPercentages (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>` from the `ccrz.ccLogicCartPriceAsync.priceItem` method. The map can include the following keys:

### `ccrz.ccLogicCartPriceAsync.PRICE_ITEM_CART_ITEM`

`ccrz__E_CartItem__c` that's a minor line item under a kit or dynamic kit.

### `ccrz.ccLogicCartPriceAsync.PRICE_ITEM_PARENT_ITEM`


The parent `ccrz__E_CartItem__c` of `ccrz.ccLogicCartPriceAsync.PRICE_ITEM_CART_ITEM`. This cart item must be a kit or dynamic kit that defines a subscription term.

### `ccrz.ccLogicCartPriceAsync.PRICE_ITEM_SUBPRODTERM_MAP`

`Map<String, Object>` that describes the `ccrz__E_SubProdTerm__c` record that the cart item references.

## Outputs

This method returns the input `Map<String, Object>` and updates the values of the following `ccrz__E_CartItem__c` fields.

 **Note:** By default, this method applies price information from the modifier subscription term with the lowest Sequence value.

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__Price__c</code>	The standard price of the cart item, adjusted by the subscription's upfront (non-recurring) percentage.
<code>ccrz__RecurringPrice__c</code>	The standard price of the cart item, adjusted by the subscription's recurring percentage.
<code>ccrz__RecurringPriceSubAmt__c</code>	The recurring price multiplied by quantity.
<code>ccrz__SubAmount__c</code>	The upfront price multiplied by quantity.

SEE ALSO:

[ccrz\\_\\_E\\_SubProdTerm\\_\\_c](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## `ccrz.ccLogicCartPriceAsync.priceAttributeItem`

Parses JSON attribute pricing data for a cart item, and sets the item's corresponding price fields. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> priceAttributeItem(Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>` from the `ccrz.ccLogicCartPriceAsync.priceItem` method. The map can include the following keys:

### `ccrz.ccApiAttribute.ATTRIBUTES`

`Map<String, ccrz__E_Attribute__c>` returned from the `ccrz.ccApiAttribute.fetch` query that the `ccrz.ccLogicCartPrice.fetchPricingData` method invoked.

### `ccrz.ccLogicCartPriceAsync.PRICE_ITEM_ATTRIBUTE_PRICING`

`Map<String, Object>` that includes the deserialized JSON attribute pricing data from the corresponding price list item.

### `ccrz.ccLogicCartPriceAsync.PRICE_ITEM_CART_ITEM`

`ccrz__E_CartItem__c` that's a minor line item under a kit or dynamic kit.

## Outputs

This method returns the input `Map<String, Object>` and updates the values of the following `ccrz__E_CartItem__c` fields.

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__Price__c</code>	The price specified by the cart item's applicable primary, secondary, or tertiary attribute.
<code>ccrz__SubAmount__c</code>	The price multiplied by quantity.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccApiAttribute.fetch](#)

[ccrz.ccLogicCartPriceAsync.fetchPricingData](#)

## `ccrz.ccLogicCartPriceAsync.checkCartItemValidity`

Continues the For loop within the `ccrz.ccLogicCartPriceAsync.applyPricing` method by evaluating validation criteria and updating the cart item's status. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> checkCartItemValidity(Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>` from the `ccrz.ccLogicCartPriceAsync.applyPricing` method. The map can include the following keys:

**ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_CART\_ITEM**

`ccrz__E_CartItem__c` to evaluate.

**ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_PARENT\_ITEM**

The parent `ccrz__E_CartItem__c` of `ccrz.ccLogicCartPriceAsync.PRICE_ITEM_CART_ITEM`.

## Outputs

This method returns the input `Map<String, Object>` and sets the value of the `ccrz__E_CartItem__c` record's `ccrz__ItemStatus__c` field to `Invalid` for any of the following conditions:

- The cart item's price is negative.
- The corresponding product's Start Date or End Date is invalid for the current date.
- The product's `ccrz__ProductStatus__c` field has a value other than `Released`, and the product isn't a component of a bundle.

When the cart item is invalid, this method also sets the following fields:

ccrz__E_CartItem__c Field	Value
<code>ccrz__OriginalItemPrice__c</code>	The same value as <code>ccrz__Price__c</code> .
<code>ccrz__Price__c</code>	-1
<code>ccrz__RecurringPrice__c</code>	null
<code>ccrz__RecurringPriceSubAmt__c</code>	null
<code>ccrz__SubAmount__c</code>	null



**Note:** If the cart item has a parent cart item, this method also sets the parent item's status to `Invalid`.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCartPriceAsync.evaluateComplexProducts

Evaluates cart items that are included in complex products and sets the status of each cart item. A *complex product* can be a bundle, kit, dynamic kit, assembly product, or attribute-based product. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> evaluateComplexProducts (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.applyPricing` method. The map can include the following keys:

#### **ccrz.ccLogicCartPriceAsync.COMPLETE\_INPUT\_DATA**

`Map<String, Object>` of the working map passed from `ccrz.ccLogicCartPriceAsync.applyPricing`.

#### **ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_CART\_ITEMS**

`Map<String, ccrz__E_CartItem__c>` of cart items that have parent cart items, or cart items that are part of attribute-based products (where `ccrz__PricingType__c` is `ATTRGROUP`).

#### **ccrz.ccLogicCartPriceAsync.PRICE\_ITEM\_COMPLEX\_CART\_ITEMS**

`Map<String, List<ccrz__E_CartItem__c>>`, keyed by parent cart item ID, that describes major cart line items and its associated minor cart line items.

### Outputs

This method returns the input `Map<String, Object>` and sets the value of the `ccrz__E_CartItem__c` record's `ccrz__ItemStatus__c` field to `Invalid` for any of the following conditions:

- The cart item is a major line item for a complex product (other than an assembly product) but it doesn't have any child cart items.
- The cart item's `ccrz__PricingType__c` is `ATTRGROUP` and contains child cart items where the `ccrz__ItemStatus__c` is already set to `Invalid`. This method also sets the major line item to `Invalid`.
- The cart item is a major line item for a kit or dynamic kit and contains child cart items where the `ccrz__ItemStatus__c` is already set to `Invalid`. This method also sets the major line item to `Invalid`.

When the cart item is invalid, this method also sets the following fields:

<code>ccrz__E_CartItem__c</code> Field	Value
<code>ccrz__OriginalItemPrice__c</code>	The same value as <code>ccrz__Price__c</code> .
<code>ccrz__Price__c</code>	-1



ccrz__E_CartItem__c <b>Field</b>	<b>Value</b>
ccrz__RecurringPrice__c	null
ccrz__RecurringPriceSubAmt__c	null
ccrz__SubAmount__c	null

For a kit or dynamic kit, this method also evaluates any subscription terms applied to the major line item, and sets the following fields:

ccrz__E_CartItem__c <b>Record</b>	<b>Field</b>	<b>Value</b>
Major line item (composite product for a kit or dynamic kit)	ccrz__RecurringPrice__c	Calculated from the child cart item's ccrz__RecurringPrice__c
	ccrz__RecurringPriceSubAmt__c	Recurring price multiplied by quantity.
Child line item (component product for a kit or dynamic kit)	ccrz__RecurringPrice__c	null
	ccrz__RecurringPriceSubAmt__c	null

SEE ALSO:

[Salesforce Help Types of Products for B2B Commerce for Visualforce](#)

## ccrz.ccLogicCartPriceAsync.performExternalPricing

Provides a placeholder for pricing a cart from an external source. By default, this method is empty and performs no default action. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> performExternalPricing (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.applyPricing` method completes.

By default, this method doesn't evaluate any input data.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

## ccrz.ccLogicCartPriceAsync.performDMLOnItems

Removes cart items, combines externally priced cart items with other cart items, and finally executes the DML `upsert` operation for the updated `ccrz__E_CartItem__c` records. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> performDMLOnItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.performExternalPricing` method completes.

This method requires the following key in the working map:

### **ccrz.ccLogicCartPriceAsync.ITEMS\_TO\_DELETE**

List<`ccrz__E_CartItem__c`> of the items to remove from the working map. The `ccrz.ccLogicCartPriceAsync.prepareCartItems` method returns this list.

### **ccrz.ccLogicCartPriceAsync.ITEMS\_TO\_PRICE**

Map<String, `ccrz__E_CartItem__c`> of the cart's items that previous methods have priced and that are ready to be upserted.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### **ccrz.ccLogicCartPriceAsync.EXTERNALLY\_PRICED\_ITEMS**

List<`ccrz__E_CartItem__c`> of the items where `ccrz__PricingType__c` is external. These items were excluded from the previous pricing methods.

## Logic Performed

This method completes the following actions:

1. Performs the DML `delete` operation on `ccrz.ccLogicCartPriceAsync.ITEMS_TO_DELETE`.
2. Performs the DML `upsert` operation on `ccrz.ccLogicCartPriceAsync.ITEMS_TO_PRICE` and `ccrz.ccLogicCartPriceAsync.EXTERNALLY_PRICED_ITEMS`.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[Deleting Records](#)

[Upserting Records](#)

## ccrz.ccLogicCartPriceAsync.enqueueNext

Determines whether more major line item `ccrz__E_CartItem__c` records remain to be priced, and calls `ccrz.ccLogicCartPriceAsync.enqueuePricing` for queuing the next job. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> enqueueNext (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.performDMLOnItems` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiCart.CARTITEMLASTID**

String that specifies the Salesforce ID of the final major cart item that was processed during the current recursive job.



**Note:** If this input isn't specified, this method doesn't execute correctly and can return an error.

### **ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to price.

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

## Logic Performed

This method completes the following actions:

1. Reads `ccrz.ccApiCart.CARTITEMLASTID` and determines where in the list of cart items the current recursive job finished processing.

- Executes a direct query of `ccrz__E_CartItem__c` records for the source `ccrz__E_Cart__c` to determine whether more `ccrz__E_CartItem__c` records remain to be processed:
  - If more `ccrz__E_CartItem__c` records remain, this method sets the `ccrz.ccApiCart.NEXT_PRICING_JOB_ENQUEUED` output to `true`.
  - Otherwise, this method sets the `ccrz.ccApiCart.NEXT_PRICING_JOB_ENQUEUED` output to `false`.
- Sets `ccrz.ccApiCart.CARTITEMLASTID` for the next recursive job that `ccrz.ccLogicCartPriceAsync.enqueuePricing` executes.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCart.NEXT_PRICING_JOB_ENQUEUED`

Boolean

Value	Usage
<code>true</code>	Another recursive <code>ccrz.ccApiCart.priceAsync</code> call is enqueued for processing the next batch of <code>ccrz__E_CartItem__c</code> records.
<code>false</code>	The current job is the final recursive job. No other <code>ccrz__E_CartItem__c</code> records remain for processing.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccLogicCartPriceAsync.enqueuePricing](#)

## `ccrz.ccLogicCartPriceAsync.applyCoupons`

Creates a cart item for a coupon, and applies the coupon's discount to the cart. This method calls the `ccrz.ccApiCoupon.apply` method, which creates the coupon cart item, applies the discount, and executes the corresponding DML operations. This method executes only during the final recursive `ccrz.ccApiCart.priceAsync` method call that processes the remaining `ccrz__E_CartItem__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> applyCoupons (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.updateCartHeader` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiCart.NEXT\_PRICING\_JOB\_ENQUEUED**

Boolean that must be `false` for this method to execute.

Value	Usage
<code>true</code>	Another asynchronous Apex job is enqueued for continuing the pricing process.
<code>false</code>	The current asynchronous Apex Job is the final job for the pricing process.

### **ccrz.ccLogicCartPriceAsync.CART\_TO\_PRICE**

`ccrz__E_Cart__c` record that represents the cart to price.

### **ccrz.ccLogicCartPriceAsync.CART\_TO\_UPDATE**

`ccrz__E_Cart__c` record that represents the new cart record that the `ccrz.ccLogicCartPriceAsync.updateCartHeader` method created.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### **ccrz.ccLogicCartPriceAsync.ITEMS\_TO\_PRICE**

`Map<String, ccrz__E_CartItem__c>` of the cart's items that previous methods have priced and that are ready to be upserted.

### **ccrz.ccLogicCartPriceAsync.EXTERNALLY\_PRICED\_ITEMS**

`List<ccrz__E_CartItem__c>` of the items where `ccrz__PricingType__c` is `external`. These items are excluded from the subsequent pricing methods.

## Logic Performed

This method completes the following actions:

1. Sets the following fields on `ccrz.ccLogicCartPriceAsync.CART_TO_PRICE` to match `ccrz.ccLogicCartPriceAsync.CART_TO_UPDATE`:
  - `ccrz__AsynchronousJobID__c`
  - `ccrz__CartStatus__c`
  - `ccrz__RepricedDate__c`
2. Combines the cart items from `ccrz.ccLogicCartPriceAsync.ITEMS_TO_PRICE` and `ccrz.ccLogicCartPriceAsync.EXTERNALLY_PRICED_ITEMS` into a single intermediate `List<ccrz__E_CartItem__c>`.
3. Passes the intermediate `List<ccrz__E_CartItem__c>` and `ccrz.ccLogicCartPriceAsync.CART_TO_PRICE` as inputs for the `ccrz.ccApiCoupon.apply` method.

## Outputs

This method returns the working `Map<String, Object>` after invoking the `ccrz.ccApiCoupon.apply` method.

SEE ALSO:

[ccrz.ccApiCoupon.apply](#)

[Salesforce Help Offer Coupons for Discounts on Storefront Purchases](#)

## ccrz.ccLogicCartPriceAsync.enqueuePlace

Calls the `System.enqueueJob` method for adding the queueable `ccLogicCartPriceEnqueuePlace` class to the asynchronous Apex job queue. To call a custom queueable class, override this method with custom subscriber code. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls, and only when pricing occurs as part of the `ccrz.ccApiCart.placeAsync` logic chain.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> enqueuePlace (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.applyCoupons` method completes.

This method requires the following key in the working map:

**ccrz.ccApiCart.CONTINUE\_ASYNC\_PLACE\_AFTER\_PRICING**

Boolean that must be `true` for this method to execute.

Value	Usage
<code>true</code>	This method is executing as part of the <code>ccrz.ccApiCart.placeAsync</code> method. When this method completes, it calls <code>ccrz.ccApiCart.placeAsync</code> recursively to continue the order placement process.
<code>false</code>	This method is executing outside of the <code>ccrz.ccApiCart.placeAsync</code> method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiCart.PLACE\_ASYNC\_JOB\_ID**

ID of the `AsyncApexJob` that represents the asynchronous Apex job that the `System.enqueueJob` method returned.

SEE ALSO:

[ccrz.ccApiCart.placeAsync](#)

[enqueueJob \(queueableObj\)](#)

[Queueable Apex](#)

## ccrz.ccLogicCartPriceAsync.updateOnError

Updates the fields on the `ccrz__E_Cart__c` record when an exception occurs during asynchronous cart pricing. This method executes only during the recursive, asynchronous `ccrz.ccApiCart.priceAsync` method calls.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> updateOnError (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartPriceAsync.executeAsyncPrice` method.

This method requires the following keys in the working map:

**ccrz.ccLogicCartPriceAsync.CART\_TO\_PRICE**

`ccrz__E_Cart__c` record that represents the cart to price.

### Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccLogicCartPriceAsync.RETRYING\_ASYNC**

Boolean that must be `true` for this method to execute.

Value	Usage
<code>true</code>	An error occurred when pricing the cart during the previous iteration of the recursive, asynchronous call. The logic chain returns a <code>ccrz.ccLogicCartPriceAsync.CartPriceAsyncException</code> .
<code>false</code>	No error occurred.

## Logic Performed

This method performs the DML `update` operation on the `ccrz__E_Cart__c` and sets values for the following fields:

Field	Value
<code>ccrz__CartStatus__c</code>	<code>Open</code>
<code>ccrz__RepricedDate__c</code>	<code>null</code>

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartPriceAsync.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys. This method executes for every `ccrz.ccApiCart.priceAsync` call, regardless of whether it's the initial or a recursive call.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartPriceAsync.executeAsyncPrice` method completes.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`:

- `ccrz.ccApiProduct.PRODUCT_LIST`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicCartPriceAsync.CART_ITEM_LIST`



- `ccrz.ccLogicCartPriceAsync.CART_TO_PRICE`
- `ccrz.ccLogicCartPriceAsync.EXTERNALLY_PRICED_ITEMS`
- `ccrz.ccLogicCartPriceAsync.ITEMS_TO_DELETE`
- `ccrz.ccLogicCartPriceAsync.ITEMS_TO_PRICE`
- `ccrz.ccLogicCartPriceAsync.ITEMS_TO_UPDATE`
- `ccrz.ccLogicCartPriceAsync.MAJORS_WITH_EXTERNAL_CHILDREN`
- `ccrz.ccLogicCartPriceAsync.PRODUCT_PRICING_DATA`

## ccrz.ccLogicCartPriceAsync.handleCartPriceException

Catches a `ccrz.ccLogicCartPriceAsync.CartPriceAsyncException` that occurs during the logic chain, and logs the exception message. If the pricing logic is executing as part of a `ccrz.ccApiCart.placeAsync` call, this method also catches a `ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException` to ensure that the error propagates and ends order placement.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> handleCartPriceException (Map<String, Object>)
```

### Inputs (Required)

`Map<String, Object>` that can include the following keys:

#### **ccrz.ccApiCart.CONTINUE\_ASYNC\_PLACE\_AFTER\_PRICING**

Boolean

Value	Usage
<code>true</code>	This method is executing as part of the <code>ccrz.ccApiCart.placeAsync</code> method. This method catches both <code>ccrz.ccLogicCartPriceAsync.CartPriceAsyncException</code> and <code>ccrz.ccLogicCartPlaceAsync.CartPlaceAsyncException</code> .
<code>false</code>	This method is executing outside of the <code>ccrz.ccApiCart.placeAsync</code> method.

#### **ccrz.ccLogicCartPriceAsync.CART\_PRICE\_EXCEPTION**

`ccrz.ccLogicCartPriceAsync.CartPriceAsyncException` that occurred during the `ccrz.ccLogicCartPriceAsync.process` method or its inner methods. The `ccrz.ccLogicCartPriceAsync.process` method rethrows the exception after it's caught, which ends processing and rolls back transactions.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz.ccLogicCartPlaceAsync.handleCartPlaceException](#)

## ccrz.ccLogicCartProcessItemGroups

Define the business logic for moving cart items between shipping groups for line-level independence (LLI) checkout.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiCart.processItemGroups](#)

## Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

[processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

[processItemGroups](#)

Determines whether the cart already contains at least shipping group, and assigns cart items to the group. When no shipping group exists, this method creates a single default group with all cart items and the account's default shipping address. This method also wraps the DML `update` statement for assigning cart items.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCartProcessItemGroups.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.processItemGroups` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartProcessItemGroups` class in the following order:

1. `processInput`
2. `processItemGroups`
3. `prepReturn`

### Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartProcessItemGroups.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartProcessItemGroups.process` method.

This method checks that the input map specifies the following keys, and returns a `ccrz.ccApi.MissingInputException` if any key is missing:

### **ccrz.ccApiCart.CART\_ENCID**

The encrypted ID of the cart.

### **ccrz.ccApiCart.MOVEDATA**

`List<Map<String, Object>>` where each `Map<String, Object>` contains the following keys that describe the cart items to move.

#### **ccrz.ccApiCart.ITEMID**

String that specifies the Salesforce ID of the `ccrz__E_CartItem__c` to move.

#### **ccrz.ccApiCart.TARGETQTY**

Decimal that specifies the quantity of the `ccrz__E_CartItem__c` to move.



**Note:** If you don't specify a quantity, this method moves the full quantity.

### **ccrz.ccApiCart.TARGETGROUP**

String that specifies the Salesforce ID of the `ccrz__E_CartItemGroup__c` to move `ccrz__E_CartItem__c` records to.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartProcessItemGroups.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicCartProcessItemGroups.processItemGroups

Determines whether the cart already contains at least shipping group, and assigns cart items to the group. When no shipping group exists, this method creates a single default group with all cart items and the account's default shipping address. This method also wraps the DML `update` statement for assigning cart items.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processItemGroups (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartProcessItemGroups.processInput` method.

This method requires the following key in the working map:

**ccrz.ccApiCart.CART\_ENCID**

The encrypted ID of the cart.

**ccrz.ccApiCart.MOVEDATA**

`List<Map<String, Object>>` where each `Map<String, Object>` contains the following keys that describe the cart items to move.

**ccrz.ccApiCart.ITEMID**

String that specifies the Salesforce ID of the `ccrz__E_CartItem__c` to move.

**ccrz.ccApiCart.TARGETQTY**

Decimal that specifies the quantity of the `ccrz__E_CartItem__c` to move.



**Note:** If you don't specify a quantity, this method moves the full quantity.

**ccrz.ccApiCart.TARGETGROUP**

String that specifies the Salesforce ID of the `ccrz__E_CartItemGroup__c` to move `ccrz__E_CartItem__c` records to.

### Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccApiCart.MERGELINES**

Boolean

Value	Usage
<i>true</i>	Combine quantities of the same SKU are combined into a single line item in the target cart item group. For usage, see <a href="https://help.salesforce.com/articleView?id=b2b_commerce_page_checkout_llj_shipping.htm">https://help.salesforce.com/articleView?id=b2b_commerce_page_checkout_llj_shipping.htm</a> .
<i>false</i> (default)	Leave quantities of the same SKU on separate line items in the target cart item group.



**Note:** For a dynamic kit, subscription, or externally priced product, this method can move quantities to another group, but can't combine quantities.

**Logic Performed**

This method completes the following actions:

1. Calls the `ccrz.ccApiCart.fetch` method and returns the specified cart.
2. Loops through the `ccrz.ccApiCart.MOVEDATA` and builds lists of the cart items to move and all the cart items in the `ccrz.ccApiCart.TARGETGROUP`.
3. Moves the quantity of cart items to the appropriate `ccrz.ccApiCart.TARGETGROUP`, depending on the following factors:
  - The quantity of the cart item in its current group
  - The value of `ccrz.ccApiCart.MERGELINES`

Cart Item Quantity	ccrz.ccApiCart.MERGELINES Value	Method Behavior
The full quantity of the cart item in the current group ( <i>bulk move</i> )	<i>false</i>	<ul style="list-style-type: none"> <li>• Moves the full quantity of the cart items to the <code>ccrz.ccApiCart.TARGETGROUP</code>.</li> <li>• Leaves quantities of the same SKU in separate line items in the target group.</li> </ul>
	<i>true</i>	<ul style="list-style-type: none"> <li>• Moves the full quantity of cart items to the <code>ccrz.ccApiCart.TARGETGROUP</code>.</li> <li>• Combines quantities of the same SKU into a single line item in the target group.</li> </ul>
A partial quantity of the cart item in the current group ( <i>split move</i> )	<i>false</i>	<ul style="list-style-type: none"> <li>• Moves the partial quantity of the cart items to the <code>ccrz.ccApiCart.TARGETGROUP</code>.</li> <li>• Leaves quantities of the same SKU in separate line items in the target group.</li> </ul>

Cart Item Quantity	ccrz.ccApiCart.MERGE LINES Value	Method Behavior
	<code>true</code>	<ul style="list-style-type: none"> <li>Moves the partial quantity of the cart items to the <code>ccrz.ccApiCart.TARGETGROUP</code>.</li> <li>Combines quantities of the same SKU into a single line item in the target group.</li> </ul>


- Executes the DML operations for inserting, upserting, or deleting cart items, if applicable.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)

[Inserting and Updating Records](#)

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

## `ccrz.ccLogicCartProcessItemGroups.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartProcessItemGroups.processItemGroups` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicCartRemove

Define the business logic for removing cart items, including a coupon, from a specified cart.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiCart.removeFrom`



## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [removeItems](#)

Removes a specified coupon code or specified cart items from the cart, and resets certain cart fields.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCartRemove.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.removeFrom` method.

## Logic Performed


This method invokes other methods of the `ccrz.ccLogicCartRemove` class in the following order:

1. `processInput`

2. `removeItems`
3. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## `ccrz.ccLogicCartRemove.processInput`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartRemove.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

**`ccrz.ccApiCart.CART_ENCID`**

String that specifies the encrypted ID of the cart to remove items from.

**`ccrz.ccApiCart.CART_ID`**

String that specifies the ID of the cart to remove items from.

This method checks that the input map also specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

**`ccrz.ccApiCart.COUPON_CODE`**

String that specifies a coupon code to remove from the cart.

**ccrz.ccApiCart.LINE\_DATA**

List<ccrz.ccApiCart.LineData> that describes the `ccrz__E_CartItem__c` records to remove from the cart. For each item, add a new `ccrz.ccApiCart.LineData` object to the list and set the `LineData.sfid` field cart item's Salesforce ID. For information about this list's keys and requirements, see [Inputs for Cart Line Data](#).



**Tip:** This key accepts only the List<ccrz.ccApiCart.LineData> representation of line data, and doesn't accept the List<Map<String, Object>> representation that we recommend for the `ccrz.ccApiCart.addTo` method.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The `ccrz.ccLogicCartRemove.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicCartRemove.removeItems

Removes a specified coupon code or specified cart items from the cart, and resets certain cart fields.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> removeItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the Map<String, Object> from the `ccrz.ccLogicCartRemove.process` method.

This method executes only if the working map includes either of the following keys:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to remove items from.

**ccrz.ccApiCart.CART\_ID**

String that specifies the ID of the cart to remove items from.

This method also requires either of the following keys:

**ccrz.ccApiCart.COUPON\_CODE**

String that specifies a coupon code to remove from the cart.

**ccrz.ccApiCart.LINE\_DATA**

List<ccrz.ccApiCart.LineData> that describes the [ccrz\\_\\_E\\_CartItem\\_\\_c](#) records to remove from the cart. For each item, add a new `ccrz.ccApiCart.LineData` object to the list and set the `LineData.sfid` field cart item's Salesforce ID. For information about this list's keys and requirements, see [Inputs for Cart Line Data](#).



**Tip:** This key accepts only the List<ccrz.ccApiCart.LineData> representation of line data, and doesn't accept the List<Map<String, Object>> representation that we recommend for the `ccrz.ccApiCart.addTo` method.

## Logic Performed

This method completes the following actions:

1. The method first checks for whether `ccrz.ccApiCart.COUPON_CODE` is specified, and invokes the non-global `ccrz.cc_api_cart.removeCoupon` method to remove the specified coupon.
2. Otherwise, the method parses `ccrz.ccApiCart.LINE_DATA` and removes the specified cart items from the cart.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart with the specified coupon code or cart items removed.

This method also sets the following fields on the [ccrz\\_\\_E\\_Cart\\_\\_c](#) record:

ccrz__E_Cart__c Field	Value
ccrz__RepricedDate__c	null
ccrz__ValidationStatus__c	null

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[Salesforce Help Offer Coupons for Discounts on Storefront Purchases](#)

## ccrz.ccLogicCartRemove.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartRemove.removeItems` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicCartRemoveCart

Define the business logic for deleting carts, including wish lists.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiCart.removeCart`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [removeCart](#)

Executes the DML `delete` operation on the cart records to remove. If a deleted cart or wish list was active for the current account, this method also determines which cart or wish list becomes active next.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCartRemoveCart.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.removeCart` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartRemoveCart` class in the following order:

1. processInput
2. removeCart
3. prepReturn

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartRemoveCart.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartRemoveCart.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to delete.

**ccrz.ccApiCart.CART\_ENCIDLIST**

Set<String> of encrypted IDs of the carts to delete.

**ccrz.ccApiCart.CART\_ID**

String that specifies the ID of a cart to delete.

**ccrz.ccApiCart.CART\_IDLIST**

Set<String> of IDs of carts to delete.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccLogic.ORIGINAL_INPUT`

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartRemoveCart.prepReturn` method removes this intermediate key from the working map.

## `ccrz.ccLogicCartRemoveCart.removeCart`

Executes the DML `delete` operation on the cart records to remove. If a deleted cart or wish list was active for the current account, this method also determines which cart or wish list becomes active next.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> removeCart (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartRemoveCart.processInput` method.

This method executes only if the working map includes one of the following keys:

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to delete.

### `ccrz.ccApiCart.CART_ENCIDLIST`

`Set<String>` of encrypted IDs of the carts to delete.

### `ccrz.ccApiCart.CART_ID`

String that specifies the ID of a cart to delete.

### `ccrz.ccApiCart.CART_IDLIST`

`Set<String>` of IDs of carts to delete.

## Logic Performed

This method completes the following actions:




1. Calls the `ccrz.ccApiCart.fetch` method to query specified carts.
2. Executes the DML `delete` operation on the returned carts.
3. If the deleted cart or wish list was active, this method also calls the `ccrz.ccApiCart.getActive` or `ccrz.ccApiWishList.getActive` method to determine which cart or wish list becomes active next.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , consider rolling back the API transaction to a previous savepoint.

### SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[ccrz.ccApiCart.getActive](#)

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)

[Deleting Records](#)

## `ccrz.ccLogicCartRemoveCart.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartRemoveCart.removeCart` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicCartRemovelItemGroups

Define the business logic for deleting shipping groups of cart items for line-level independence (LLI) checkout.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiCart.removeCartItemGroup](#)

## Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

[processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

[deleteItemGroups](#)

Prepares the cart item group records for deletion.

[performDML](#)

Executes the DML `delete` operation on the cart item group records to remove.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## SEE ALSO:

[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

## ccrz.ccLogicCartRemoveItemGroups.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.removeCartItemGroup` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartRemoveItemGroups` class in the following order:

1. `processInput`
2. `deleteItemGroups`, which calls into `performDML`
3. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartRemoveItemGroups.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartRemoveItemGroups.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

### **ccrz.ccApiCart.ITEMGROUPLIST**

`List<ccrz__E_CartItemGroup__c>` of the cart item groups to delete.

### **ccrz.ccService.TRANSFORMINDEX**

`Map<String, Object>` that the `ccrz.ccServiceCart` data service provider generates for defining how the fields in a map convert to sObject data.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicCartRemoveItemGroups.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicCartRemoveItemGroups.deleteItemGroups

Prepares the cart item group records for deletion.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> deleteItemGroups (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCartRemoveItemGroups.processInput` method completes.

This method requires the following keys in the working map:

#### **ccrz.ccApiCart.ITEMGROUPLIST**

`List<ccrz__E_CartItemGroup__c>` of the cart item groups to delete.

#### **ccrz.ccService.TRANSFORMINDEX**

`Map<String, Object>` that the `ccrz.ccServiceCart` data service provider generates for defining how the fields in a map convert to sObject data.

### Logic Performed

This method completes the following actions:

1. Parses `ccrz.ccApiCart.ITEMGROUPLIST` and `ccrz.ccService.TRANSFORMINDEX`, and transforms fields to sObject names, such as `groupName` to `ccrz__GroupName__c`.
2. Serializes the transformed data back to a `List<ccrz__E_CartItemGroup__c>` in the 'itemGroupsToDelete' output.
3. Calls the `performDML` inner method for executing the DML `delete` operation.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The method completed, and the <code>performDML</code> inner method completed successfully.
false	The <code>'itemGroupsToDelete'</code> output is null or empty.

**'itemGroupsToDelete'**List<[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)> of the cart item groups to delete.This method also removes the `ccrz.ccApiCart.ITEMGROUPLIST` input from the working map.

SEE ALSO:

[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)**ccrz.ccLogicCartRemovelItemGroups.performDML**Executes the DML `delete` operation on the cart item group records to remove.**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> performDML (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the `Map<String, Object>` from the [ccrz.ccLogicCartRemoveItemGroups.deleteItemGroups](#) method.

This method requires the following key in the working map:

**'itemGroupsToDelete'**List<[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)> of the cart item groups to delete.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[Deleting Records](#)

## ccrz.ccLogicCartRemoveItemGroups.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartRemoveItemGroups.deleteItemGroups` method completes.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApi.SIZING`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `'itemGroupsToDelete'`

## ccrz.ccLogicCartRepair

Define the business logic for reopening a cart where the status remains Repricing even after a repricing job completes. A cart can remain in this status when a repricing job is unsuccessful or no longer available in the org.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Global API Caller

`ccrz.ccApiCart.repair`

## Methods

### IN THIS SECTION:

#### `process`

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

#### `processInput`

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

#### `repair`

Calls the `ccrz.ccApiCart.fetch` method for returning carts that are candidates for repairs, evaluates whether to repair them, and updates appropriate cart field values.

#### `performDML`

Executes the DML `update` operation on the `ccrz__E_Cart__c` records.

#### `refetch`

Calls the `ccrz.ccApiCart.fetch` method for fetching the cart records after they're updated.

#### `prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## `ccrz.ccLogicCartRepair.process`

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12



## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.repair` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartRepair` class in the following order:

1. `processInput`
2. `repair`
3. `performDML`
4. `refetch`
5. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.



**Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartRepair.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartRepair.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to repair.

**ccrz.ccApiCart.CART\_ENCIDLIST**

Set<String> of encrypted IDs of the carts to repair.

**ccrz.ccApiCart.CART\_ID**

String that specifies the ID of a cart to repair.

**ccrz.ccApiCart.CART\_IDLIST**

Set<String> of IDs of carts to repair.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The `ccrz.ccLogicCartRepair.prepReturn` method removes this intermediate key from the working map.

This method also instantiates the following return data keys:

**ccrz.ccApi.SUCCESS**

Boolean that's assigned an intermediate value of `true`. Subsequent methods in the logic chain update this value.

## ccrz.ccLogicCartRepair.repair

Calls the `ccrz.ccApiCart.fetch` method for returning carts that are candidates for repairs, evaluates whether to repair them, and updates appropriate cart field values.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> repair(Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCartRepair.processInput` method completes.

This method requires one of the following keys in the working map:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to repair.

**ccrz.ccApiCart.CART\_ENCIDLIST**

Set<String> of encrypted IDs of the carts to repair.

**ccrz.ccApiCart.CART\_ID**

String that specifies the ID of a cart to repair.

**ccrz.ccApiCart.CART\_IDLIST**

Set<String> of IDs of carts to repair.

## Logic Performed

This method completes the following actions:

1. Passes `ccrz.ccApiCart.CART_ID`, `ccrz.ccApiCart.CART_IDLIST`, `ccrz.ccApiCart.CART_ENCID`, or `ccrz.ccApiCart.CART_ENCIDLIST` as input for the `ccrz.ccApiCart.fetch` method, which queries for `ccrz__E_Cart__c` records where the `ccrz__CartStatus__c` field's value is Repricing.
2. Adds each returned `ccrz__E_Cart__c` record's `ccrz__AsynchronousJobID__c` to a list of job IDs.
3. Executes a SOQL query of `AsyncApexJob` records to find asynchronous Apex jobs that match the IDs.
4. Determines which `ccrz__E_Cart__c` records require repair:
  - If a `ccrz__E_Cart__c` record has a null value for `ccrz__AsynchronousJobID__c`, the cart requires repair.
  - If a `ccrz__E_Cart__c` is assigned an `ccrz__AsynchronousJobID__c`, and the job's status is `Completed`, `Failed`, or `Aborted`, the cart requires repair.
  - If a `ccrz__E_Cart__c` is assigned an `ccrz__AsynchronousJobID__c`, and the job's status is `Processing`, `Holding`, or `Queued`, the cart's `Repricing` status is considered accurate. The cart doesn't require repair.
5. Adds each cart that requires repair to a list, and updates the following fields on each `ccrz__E_Cart__c` record:

Field	Value
<code>ccrz__CartStatus__c</code>	Open
<code>ccrz__RepricedDate__c</code>	Null

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

The `ccrz.ccLogicCartRepair.prepReturn` method removes this intermediate key from the working map.

**ccrz.ccLogicCartRepair.CARTS\_TO\_UPDATE**

List<`ccrz__E_Cart__c`> that contains the repaired carts to update.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz.ccApiCart.fetch](#)

[AsyncApexJob](#)

## ccrz.ccLogicCartRepair.performDML

Executes the DML `update` operation on the `ccrz__E_Cart__c` records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> performDML (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCartRepair.performDML` method completes.

This method requires the following key in the working map:

**ccrz.ccLogicCartRepair.CARTS\_TO\_UPDATE**


`List<ccrz__E_Cart__c>` that contains the repaired carts to update.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

SEE ALSO:

[Inserting and Updating Records](#)

## ccrz.ccLogicCartRepair.refetch

Calls the `ccrz.ccApiCart.fetch` method for fetching the cart records after they're updated.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)


This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartRepair.performDML` method completes.

This method requires the following key in the working map:

### `ccrz.ccApi.SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCart.CART_OBJLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 **Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCartRepair.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCartRepair.refetch` method completes.

### Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicCartRepair.CARTS_TO_UPDATE`

## ccrz.ccLogicCartSave

Define the business logic for saving a cart, including invoking validation logic and updating cart fields for a selected effective account.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

### Global API Caller

`ccrz.ccApiCart.saveCart`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [retrieveCartData](#)

Calls `ccrz.ccApiCart.fetch` to return the cart to evaluate for saving.

#### [externalPriceValidation](#)

Provides a placeholder for modifying cart item prices before saving the cart, such as calling and applying price information from an external source. For a composite product such as a kit, make sure to recalculate the composite product price after applying external prices. By default, this method is empty and performs no default action.

#### [updateBeforeSave](#)

Provides a placeholder for evaluating specific criteria before saving the cart, such as checking product quantity rules or enforcing other requirements. By default, this method is empty and performs no default action.

#### [updateCartAddresses](#)

Calls the `ccrz.cc_hk_EffectiveAccount.saveCart` method for evaluating a selected effective account, if enabled, and updating the cart's shipping and billing addresses.

#### [validateCart](#)

Calls the `ccrz.ccApiCart.validate` method before saving the cart.

#### [performCartSave](#)

Evaluates whether the buyer is allowed to check out from the Shopping Cart page, and updates the cart and cart items to save any changes. This method also calls the `ccrz.ccLogicCartSave.updateCartAddresses` method for evaluating a selected effective account, if enabled, and updating the cart's shipping and billing addresses.

#### [refetch](#)

Calls the `ccrz.ccApiCart.fetch` method for fetching the saved cart record to include in the global API return data.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## `ccrz.ccLogicCartSave.process`

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.saveCart` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartSave` class in the following order:

1. `processInput`
2. `retrieveCartData`
3. `externalPriceValidation`
4. `validateCart`
5. `updateBeforeSave`
6. `updateCartAddresses`
7. `validateCart`
8. `performCartSave`
9. `refetch`
10. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## `ccrz.ccLogicCartSave.processInput`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartSave.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

### **ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to save.

### **ccrz.ccApiCart.CART\_ID**

String that specifies the ID of a cart to save.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

### **ccrz.ccLogicCartSave.FETCHINPUTMAP**

`Map<String, Object>` of query input data to pass to the `ccrz.ccLogicCartSave.retrieveCartData` method.

## ccrz.ccLogicCartSave.retrieveCartData

Calls `ccrz.ccApiCart.fetch` to return the cart to evaluate for saving.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

## Signature

```
global virtual Map<String, Object> retrieveCartData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartSave.processInput` method.

This method requires the following key in the working map:

### **ccrz.ccLogicCartSave.FETCHINPUTMAP**

`Map<String, Object>` of query input data for the `ccrz.ccApiCart.fetch` method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the fetched cart to evaluate for saving.

### **ccrz.ccApiCart.CART\_ID**

String that specifies the ID of the fetched cart to evaluate for saving.

### **ccrz.ccLogicCartSave.CURRENT\_CART**

`ccrz__E_Cart__c` that represents the fetched cart to evaluate for saving.

### **ccrz.ccLogicCartSave.CURRENT\_CARTITEMS**

List<`ccrz__E_CartItem__c`> of the fetched cart's items, keyed by cart ID.



**Note:** If the `ccrz.ccApiCart.fetch` method can't return a cart that matches the `ccrz.ccLogicCartSave.FETCHINPUTMAP` query, or if the cart contains no items, this method returns a `ccrz.ccLogicCartSave.CartSaveException`.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCartSave.externalPriceValidation

Provides a placeholder for modifying cart item prices before saving the cart, such as calling and applying price information from an external source. For a composite product such as a kit, make sure to recalculate the composite product price after applying external prices. By default, this method is empty and performs no default action.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

## Signature

```
global virtual Map<String, Object> externalPriceValidation (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartSave.retrieveCartData` method. By default, this method doesn't evaluate any input data.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

## ccrz.ccLogicCartSave.updateBeforeSave

Provides a placeholder for evaluating specific criteria before saving the cart, such as checking product quantity rules or enforcing other requirements. By default, this method is empty and performs no default action.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

## Signature

```
global virtual Map<String, Object> updateBeforeSave (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartSave.externalPriceValidation` method.

By default, this method doesn't evaluate any input data.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

## ccrz.ccLogicCartSave.updateCartAddresses

Calls the `ccrz.cc_hk_EffectiveAccount.saveCart` method for evaluating a selected effective account, if enabled, and updating the cart's shipping and billing addresses.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

## Signature

```
global virtual Map<String, Object> updateCartAddresses (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartSave.updateBeforeSave` method.

This method requires the following key in the working map:

### **ccrz.ccLogicCartSave.CURRENT\_CART**

`ccrz__E_Cart__c` that represents the cart to evaluate for saving.

This method passes `ccrz.ccLogicCartSave.CURRENT_CART` as the `ccrz.cc_hk_EffectiveAccount.PARAM_CART_DATA` input for the `ccrz.cc_hk_EffectiveAccount.saveCart` method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccLogicCartSave.CURRENT\_CART**

`ccrz__E_Cart__c` that represents the cart to evaluate for saving, updated with the following values:

Field Name	Updated Value
<code>ccrz__BillTo__c</code>	A clone of the effective account's default billing address.
<code>ccrz__EffectiveAccountID__c</code>	ID of the account record that corresponds to the selected effective account.

Field Name	Updated Value
ccrz__RequestDate__c	The selected or default delivery date returned from the <a href="#">ccrz.cc_api_DeliveryDate</a> extension point class, if applicable.
ccrz__ShipTo__c	A clone of the effective account's default shipping address.

SEE ALSO:

[ccrz.cc\\_hk\\_EffectiveAccount.saveCart](#)

## ccrz.ccLogicCartSave.validateCart

Calls the `ccrz.ccApiCart.validate` method before saving the cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

### Signature

```
global virtual Map<String, Object> validateCart (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the [ccrz.ccLogicCartSave.updateCartAddresses](#) method.

This method requires the following key in the working map:

**ccrz.ccLogicCartSave.CURRENT\_CART**  
[ccrz\\_\\_E\\_Cart\\_\\_c](#) that represents the cart to evaluate for saving.

This method passes the cart ID of `ccrz.ccLogicCartSave.CURRENT_CART` as the `ccrz.ccApiCart.CART_ID` input for the [ccrz.ccApiCart.validate](#) method.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApi.MESSAGES**  
`List<ccrz.cc_bean_Message>`, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

**ccrz.ccApiCart.ALLOW\_CHECKOUT**

Boolean

Value	Usage
<code>true</code>	The buyer is allowed to check out from the Shopping Cart page.
<code>false</code>	The buyer isn't allowed to check out from the Shopping Cart page.

SEE ALSO:

[ccrz.ccApiCart.validate](#)**ccrz.ccLogicCartSave.performCartSave**

Evaluates whether the buyer is allowed to check out from the Shopping Cart page, and updates the cart and cart items to save any changes. This method also calls the `ccrz.ccLogicCartSave.updateCartAddresses` method for evaluating a selected effective account, if enabled, and updating the cart's shipping and billing addresses.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

**Signature**

```
global virtual Map<String, Object> performCartSave (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartSave.validateCart` method.

This method requires the following keys in the working map:

**ccrz.ccApiCart.ALLOW\_CHECKOUT**

Boolean

Value	Usage
<code>true</code>	The buyer is allowed to check out from the Shopping Cart page.
<code>false</code>	The buyer isn't allowed to check out from the Shopping Cart page.

**ccrz.ccLogicCartSave.CURRENT\_CART**


`ccrz__E_Cart__c` that represents the cart to evaluate for saving.

**ccrz.ccLogicCartSave.CURRENT\_CARTITEMS**

List<`ccrz__E_CartItem__c`> of the fetched cart's items, keyed by cart ID.

## Outputs

This method returns the working `Map<String, Object>`, and updates the value of the `ccrz__ValidationStatus__c` field on the `ccrz__E_Cart__c` record.

 **Note:** If the cart's `ccrz__ActiveCart__c` field is `false`, this method also calls the `ccrz.ccApiCart.setActive` method.

SEE ALSO:

[ccrz.ccLogicCartSave.updateCartAddresses](#)

[ccrz.ccApiCart.setActive](#)

## ccrz.ccLogicCartSave.refetch

Calls the `ccrz.ccApiCart.fetch` method for fetching the saved cart record to include in the global API return data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartSave.performCartSave` method.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the fetched cart to evaluate for saving.

**ccrz.ccApiCart.CART\_ID**

String that specifies the ID of the fetched cart to evaluate for saving.


This method applies these parameters to a [ccrz.ccApiCart.fetch](#) query for returning the saved cart.

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiCart.fetch` outputs, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiCart.CART\_OBJLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents an updated `ccrz__E_Cart__c` record that the [ccrz.ccApiCart.fetch](#) method returned.

 **Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCartSave.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartSave.refetch` method.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApiCart.ALLOW_CHECKOUT`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicCartSave.CURRENT_CART`
- `ccrz.ccLogicCartSave.CURRENT_CARTITEMS`
- `ccrz.ccLogicCartSave.FETCHINPUTMAP`

## ccrz.ccLogicCartSetActive

Define the business logic for setting a single cart as the active cart for a particular owner, and updating all the owner's other open carts to inactive.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiCart.SetActive](#)

### Methods

#### IN THIS SECTION:

##### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

##### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

##### [updateActive](#)

Sets a single cart as the active cart for a particular owner, and updates all the owner's other open carts to inactive.

##### [refetch](#)

Calls the `ccrz.ccApiCart.fetch` method for fetching the active cart record to include in the global API return data.

##### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### ccrz.ccLogicCartSetActive.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.setActive` method.

## Logic Performed

This method first checks the value of `ccrz.cc_CallContext.isGuest`. When the value is `false`, this method invokes other methods of the `ccrz.ccLogicCartSetActive` class in the following order:

1. `processInput`
2. `updateActive`
3. `refetch`
4. `prepReturn`

When `ccrz.cc_CallContext.isGuest` is `true`, this method exits and returns `ccrz.ccApi.SUCCESS` with a value of `false`.

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. The method adds the following intermediate key to this map:

### `ccrz.ccApiCart.RESPONSE`

`Map<String, Object>` that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## `ccrz.ccLogicCartSetActive.processInput`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartSetActive.process` method.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### **ccrz.ccApiCart.BYEFFECTIVEACCOUNT**

Boolean

Value	Usage
<code>true</code>	Filter the cart query by the selected effective account. This method evaluates this key only when your storefront uses the <a href="#">Cart per Account mode for effective accounts</a> .
<code>false</code> (default)	Ignore effective accounts.

### **ccrz.ccApiCart.BYOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.



#### Note:

- Sharing rules in your org can affect what data you can see when you filter by cart owner.
- If you don't include this key, this method falls back to `ccrz.cc_CallContext.currUserId`.

### **ccrz.ccApiCart.BYSTOREFRONT**

String that specifies the name of a storefront for filtering the query.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### **ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to make active.

### **ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to make active.



**Note:** If you don't specify `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID`, this method falls back to `ccrz.cc_CallContext.currCartId`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccLogicCartSetActive.CARTIDLIST**

`Set<String>` of Salesforce IDs of the carts to query.

### **ccrz.ccLogicCartSetActive.CARTENCIDLIST**

`Set<String>` of encrypted IDs of the carts to query.

 Note:

- The `ccrz.ccLogicCartSetActive.CARTIDLIST` or `ccrz.ccLogicCartSetActive.CARTENCIDLIST` set is supposed to contain only one ID, but is established for the subsequent `ccrz.ccApiCart.fetch` query invoked from `ccrz.ccLogicCartSetActive.updateActive`.
- Don't combine `ccrz.ccLogicCartSetActive.CARTIDLIST` and `ccrz.ccLogicCartSetActive.CARTENCIDLIST`.
- Always return one of these sets instead of passing through `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID`, respectively.

**ccrz.ccLogicCartSetActive.CARTOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

**ccrz.ccLogicCartSetActive.EFFECTIVEACCOUNT**

String that specifies the ID of the selected effective account for the cart query. This method sets this value from `ccrz.cc_CallContext.effAccountId`.

**ccrz.ccLogicCartSetActive.STOREFRONT**

String that specifies the storefront for the cart query.

## ccrz.ccLogicCartSetActive.updateActive

Sets a single cart as the active cart for a particular owner, and updates all the owner's other open carts to inactive.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> updateActive (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartSetActive.processInput` method.

This method executes only if the working map includes either of the following keys:

**ccrz.ccLogicCartSetActive.CARTIDLIST**

`Set<String>` of Salesforce IDs of the carts to query.

**ccrz.ccLogicCartSetActive.CARTENCIDLIST**

Set<String> of encrypted IDs of the carts to query.

 **Note:**

- The `ccrz.ccLogicCartSetActive.CARTIDLIST` or `ccrz.ccLogicCartSetActive.CARTENCIDLIST` set is supposed to contain only one ID, but is established for the subsequent `ccrz.ccApiCart.fetch` query that this method invokes.
- Don't combine `ccrz.ccLogicCartSetActive.CARTIDLIST` and `ccrz.ccLogicCartSetActive.CARTENCIDLIST`.

This method also requires the following keys:

**ccrz.ccApiCart.RESPONSE**

Map<String, Object> that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

**ccrz.ccLogicCartSetActive.CARTOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

**ccrz.ccLogicCartSetActive.EFFECTIVEACCOUNT**

String that specifies the ID of the selected effective account for the cart query. This method sets this value from `ccrz.cc_CallContext.effAccountId`.

**ccrz.ccLogicCartSetActive.STOREFRONT**

String that specifies the storefront for the cart query.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccLogicCartSetActive.EFFECTIVEACCOUNT**

String that specifies the ID of the selected effective account for the cart query. This method evaluates this key only when your storefront uses the [Cart per Account mode for effective accounts](#).

## Logic Performed

This method completes the following actions:


1. Checks the value of `ccrz.cc_CallContext.isGuest`. If the value is `false`, the method executes.
2. Applies the input keys to a `ccrz.ccApiCart.fetch` query, which returns the cart specified by `ccrz.ccLogicCartSetActive.CARTIDLIST` or `ccrz.ccLogicCartSetActive.CARTENCIDLIST`.
3. Sets the `ccrz__ActiveCart__c` field to `true` on the first cart that `ccrz.ccApiCart.fetch` returns.
4. Calls `ccrz.ccApiCart.fetch` again for any other carts that match the input query criteria, and sets each cart's `ccrz__ActiveCart__c` field to `false`.

## Outputs

This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, consider rolling back the API transaction to a previous savepoint.</p>

**ccrz.ccLogicCartSetActive.CARTID\_ACTIVE**

String that specifies the Salesforce ID of the cart that was set to active.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[Manage Carts for Effective Accounts](#)

**ccrz.ccLogicCartSetActive.refetch**

Calls the `ccrz.ccApiCart.fetch` method for fetching the active cart record to include in the global API return data.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

**Inputs (Required)**


This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartSetActive.updateActive` method.

This method requires the following keys in the working map:

**ccrz.ccApi.SIZING**

Map<String, Object> that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

**ccrz.ccApiCart.RESPONSE**

Map<String, Object> that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

**ccrz.ccLogicCartSetActive.CARTID\_ACTIVE**

String that specifies the Salesforce ID of the active cart to query and include in the return data.

**ccrz.ccLogicCartSetActive.CARTOWNER**

String that specifies the ID of a specific user. Return only cart records where the Owner field matches this user ID.

**ccrz.ccLogicCartSetActive.STOREFRONT**

String that specifies the storefront for the cart query.

This method applies these parameters to a `ccrz.ccApiCart.fetch` query for returning the active cart.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccLogicCartSetActive.EFFECTIVEACCOUNT**


String that specifies the ID of the selected effective account for the cart.

## Outputs

This method returns the working Map<String, Object>, which includes the `ccrz.ccApiCart.fetch` outputs, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .



**ccrz.ccApiCart.CART\_OBJLIST**

List<Map<String, Object>>, where each Map<String, Object> represents an updated `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

 **Note:**

- This key is returned only when the input map specified `ccrz.ccApi.SZ_REFETCH => true`.
- The `ccrz.ccApiCart.fetch` call can also return `ccrz.ccApiAddress.ADDRESSLIST` and `ccrz.ccApiProduct.PRODUCTLIST` keys. For more information about returning address and product data for a cart, see [ccrz.ccApiCart.fetch](#).

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCartSetActive.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working Map<String, Object> from the `ccrz.ccLogicCartSetActive.refetch` method.

This method requires the following key in the working map:

**ccrz.ccApiCart.RESPONSE**

Map<String, Object> that describes the status of the logic chain for this method, such as the current value of the `ccrz.ccApi.SUCCESS` return key.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApiCart.RESPONSE`
- `ccrz.ccLogicCartSetActive.CARTENCIDLIST`
- `ccrz.ccLogicCartSetActive.CARTID_ACTIVE`
- `ccrz.ccLogicCartSetActive.CARTIDLIST`
- `ccrz.ccLogicCartSetActive.CARTOWNER`
- `ccrz.ccLogicCartSetActive.EFFECTIVEACCOUNT`
- `ccrz.ccLogicCartSetActive.STOREFRONT`

## ccrz.ccLogicCartTransfer

Define the business logic for transferring ownership of a cart to another user.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiCart.transferCart](#)

## Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

[processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

[transferCart](#)

Prepares the cart data for transferring to a new owner, but doesn't commit the changes yet.

### [updateCart](#)

Executes the DML `update` operation on the cart records that the `ccrz.ccLogicCartTransfer.transferCart` method evaluated.

### [updateContactAddress](#)

Executes the DML `update` operation on the contact address records related to the carts that the `ccrz.ccLogicCartTransfer.transferCart` method evaluated.

### [refetch](#)

Calls the `ccrz.ccApiCart.fetch` method for fetching the transferred and updated cart record to include in the global API return data.

### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCartTransfer.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.transferCart` method.

### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartRemoveCart` class in the following order:

1. `processInput`
2. `transferCart`

3. `updateCart`
4. `updateContactAddress`
5. `refetch`
6. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## `ccrz.ccLogicCartTransfer.processInput`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartTransfer.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to transfer.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

### `ccrz.ccApiUser.USER_ID`

String that specifies the Salesforce ID of the user to transfer ownership to.

**ccrz.ccApiUser.USER\_NAME**

String that specifies the username of the user to transfer ownership to.

**! Important:**

- By default, this method is invoked in a context where the current user is the cart's current Owner. If the cart was originally used by a guest user, the current Owner is the [default owner for all records created by guest users](#).
- To successfully transfer ownership to another user, specify the ID or username of a user that has row-level access to the cart. Otherwise, this method doesn't return any error when ownership doesn't successfully transfer.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The [ccrz.ccLogicCartTransfer.prepReturn](#) method removes this intermediate key from the working map.

## ccrz.ccLogicCartTransfer.transferCart

Prepares the cart data for transferring to a new owner, but doesn't commit the changes yet.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> transferCart (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the [ccrz.ccLogicCartTransfer.processInput](#) method.


This method requires the following keys in the working map:

**ccrz.ccApiAccount.ID**

String that specifies the Salesforce ID of the account to transfer the cart to.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to transfer.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.MissingInputException`.

#### `ccrz.ccApiCart.WAS_ANONYMOUS`

Boolean

Value	Usage
<code>true</code>	The cart was originally used by a guest user.
<code>false</code> (default)	The cart was originally used by an authenticated user.

#### `ccrz.ccApiUser.USER_ID`

String that specifies the Salesforce ID of the user to transfer ownership to.

#### `ccrz.ccApiUser.USER_NAME`

String that specifies the username of the user to transfer ownership to.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

#### `ccrz.ccApiCart.CARTCONTADDR_TO_UPSERT`

List<`ccrz__E_ContactAddr__c`> of billing and shipping addresses to set on the cart.

#### `ccrz.ccApiCart.CART_OBJLIST`

List<`ccrz__E_Cart__c`> that includes the cart to transfer. This list can potentially include other open carts that the new user currently owns, but that must change to inactive after the transfer completes.

#### `ccrz.ccApiCart.CARTITEMS_TO_UPSERT`

List<`ccrz__E_CartItem__c`> of items to include in the cart.

#### `ccrz.ccApiCart.CURRCODE`

String that specifies an ISO 4217 currency code, such as USD or JPY, to set on the cart.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## `ccrz.ccLogicCartTransfer.updateCart`

Executes the DML `update` operation on the cart records that the `ccrz.ccLogicCartTransfer.transferCart` method evaluated.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> updateCart (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartTransfer.transferCart` method.

This method requires the following keys in the working map:

### `ccrz.ccApiCart.CARTCONTADDR_TO_UPSERT`

List<`ccrz__E_ContactAddr__c`> of billing and shipping addresses to set on the cart.

### `ccrz.ccApiCart.CART_OBJLIST`

List<`ccrz__E_Cart__c`> that includes the carts to update.

### `ccrz.ccApiCart.CARTITEMS_TO_UPSERT`


List<`ccrz__E_CartItem__c`> of cart items to update.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:


### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

## Example: Example

Pass the cart and cart items through a custom validation class before the update completes successfully.

 **Note:** This example is only a partial demonstration, and requires more code than what's shown to function correctly.

```
global with sharing class customLogicCartTransfer extends ccrz.ccLogicCartTransfer {
    global virtual override Map<String, Object> updateCart (Map<String, Object>


```

```

        // Create an instance of your custom validation class, whose code isn't shown
        here
        customValidationClass validator = new customValidationClass;

        // Execute validation. In this example, the validation class takes the carts
        as the first parameter and cart items as the second parameter

if(validator.isValid(List<ccrz__E_Cart__c>) inputData.get(ccrz.ccApiCart.CART_OBJLIST),
(List<ccrz__E_CartItem__c>) inputData.get(ccrz.ccApiCart.CARTITEMS_TO_UPSERT)) {
    // execute the default ccrz.ccLogicCartTransfer.updateCart logic
    super.updateCart(inputData);
} else {
    // Return some failure notification
    inputData.put(ccrz.ccApi.SUCCESS, false);
}
return inputData;
}
}

```

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartTransfer.updateContactAddress

Executes the DML `update` operation on the contact address records related to the carts that the `ccrz.ccLogicCartTransfer.transferCart` method evaluated.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> updateContactAddress (Map<String, Object>)
```



## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartTransfer.updateCart` method.

This method requires the following key in the working map:

**ccrz.ccApiCart.CARTCONTADDR\_TO\_UPSERT**


`List<ccrz__E_ContactAddr__c>` of billing and shipping addresses to set on the cart.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicCartTransfer.refetch

Calls the `ccrz.ccApiCart.fetch` method for fetching the transferred and updated cart record to include in the global API return data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartTransfer.updateContactAddress` method.

This method requires the following keys in the working map:

### `ccrz.ccApi.SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```



**Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart to query.

## Outputs

This method returns the working `Map<String, Object>`, updated with the results of the `ccrz.ccApiCart.fetch` call.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## `ccrz.ccLogicCartTransfer.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartTransfer.refetch` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicCartValidate

Define the business logic for determining whether a buyer is allowed to check out from the Shopping Cart page. This class also replaces the legacy `ccrz.cc_api_CartExtension.allowCheckout` and `ccrz.cc_api_CartExtension.getCartMessages` methods.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiCart.validate`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

[processValidate](#)

Calls the `ccrz.ccApiCart.fetch` method to query a cart and evaluate whether the buyer is allowed to check out from the Shopping Cart page.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

[ccrz.ccLogicCartValidate7Hk Example Extension Class](#)

The managed package provides an example extension class, `ccrz.ccLogicCartValidate7Hk`, that you can use as a starting point for extending `ccrz.ccLogicCartValidate` to reference legacy methods of the `ccrz.cc_api_CartExtension` class.

## ccrz.ccLogicCartValidate.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCart.validate` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCartValidate` class in the following order:

1. `processInput`
2. `processValidate`
3. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCartValidate.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCartValidate.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted cart ID to validate.

**ccrz.ccApiCart.CART\_ID**

String that specifies the cart ID to validate.

## Inputs (Optional)

Because this logic service provider invokes the `ccrz.ccApiCart.fetch` method, the input map can also include that method's accepted inputs.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The `ccrz.ccLogicCartValidate.prepReturn` method removes this intermediate key from the working map.

This method also instantiates the following return data keys:

**ccrz.ccApi.MESSAGES**

List<ccrz.cc\_bean\_Message>, which is initially empty.

**ccrz.ccApiCart.ALLOW\_CHECKOUT**

Boolean with an initial value of `false`, which indicates that the buyer isn't allowed to check out from the Shopping Cart page.

## ccrz.ccLogicCartValidate.processValidate

Calls the `ccrz.ccApiCart.fetch` method to query a cart and evaluate whether the buyer is allowed to check out from the Shopping Cart page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processValidate(Map<String, Object>)
```

### Inputs (Required)

This method accepts the Map<String, Object> from the `ccrz.ccLogicCartValidate.processInput` method.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted cart ID to validate.

**ccrz.ccApiCart.CART\_ID**

String that specifies the cart ID to validate.

 **Note:** Include `ccrz.ccApiCart.CART_ENCID` or `ccrz.ccApiCart.CART_ID`, but not both.

### Inputs (Optional)

The working map can also include the inputs that are available for the `ccrz.ccApiCart.fetch` method.

For the `ccrz.ccApi.API_SIZING` key, this method sets the following default [data size request](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_ASSC => false,
        ccrz.ccApi.SZ_REL => new List<String>
    }
}
```

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiCart.fetch` outputs, and sets the following keys:

### `ccrz.ccApi.MESSAGES`

`List<ccrz.cc_bean_Message>`, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

### `ccrz.ccApiCart.ALLOW_CHECKOUT`

Boolean

Value	Usage
true	The buyer is allowed to check out from the Shopping Cart page.
false	The buyer isn't allowed to check out from the Shopping Cart page.



**Note:** In the default implementation of this method, this value is `true` when the `ccrz.ccApiCart.fetch` method's `ccrz.ccApiCart.CART_OBJLIST` return data key contains a cart that matches the fetch query.



### **Example: Allow Checkout for Only an Odd Number of Cart Items**

Override the default `ccrz.ccLogicCartValidate.processValidate` method and set `ccrz.ccApiCart.ALLOW_CHECKOUT` to `true` only when the total count (but not quantity) of fetched cart items is an odd number.

```
global with sharing class allowOnlyOddCartItemCount extends ccrz.ccLogicCartValidate
{
    global virtual override Map<String, Object> processValidate(Map<String, Object>
) {
        Map<String, Object> cartFetchQuery = new Map<String, Object>(inputData);
        cartFetchQuery.put(ccrz.ccApi.SIZING, new Map<String, Object> {
            ccrz.ccApi.ENTITYNAME => new Map<String, Object> {
                ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
                ccrz.ccApi.SZ_ASSC => false,
                ccrz.ccApi.SZ_SKIPTRZ => false
            }
        });
        Map<String, Object> cartFetchData = ccrz.ccApiCart.fetch(cartFetchQuery);
        Integer cartItemCount = 0;
        for(Map<String, Object> fetchedCart : (List<Map<String,
Object>>)cartFetchData.get(ccrz.ccApiCart.CART_OBJLIST)) {
            List<Object> fetchedCartItems = (List<Object>)fetchedCart.get('ECartItemsS');
```

```

        if (null != fetchedCartItems) {
            cartItemCount += fetchedCartItems.size;
        }
    }
    inputData.put(ccrz.ccApiCart.ALLOW_CHECKOUT, (0 != Math.mod(cartItemCount,
2)));
    inputData.put('cartItemCount', cartItemCount);
    inputData.putAll(ccrz.ccApiCart.fetch(inputData));
    return inputData;
}
}

```

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

[ccrz.cc\\_api\\_CartExtension.allowCheckout](#)

[ccrz.cc\\_api\\_CartExtension.getCartMessages](#)

## ccrz.ccLogicCartValidate.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCartValidate.processValidate` method.



## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicCartValidate7Hk Example Extension Class

The managed package provides an example extension class, `ccrz.ccLogicCartValidate7Hk`, that you can use as a starting point for extending `ccrz.ccLogicCartValidate` to reference legacy methods of the `ccrz.cc_api_CartExtension` class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

This extension overrides the default `ccrz.ccLogicCartValidate.processValidate` method to call the following legacy methods:

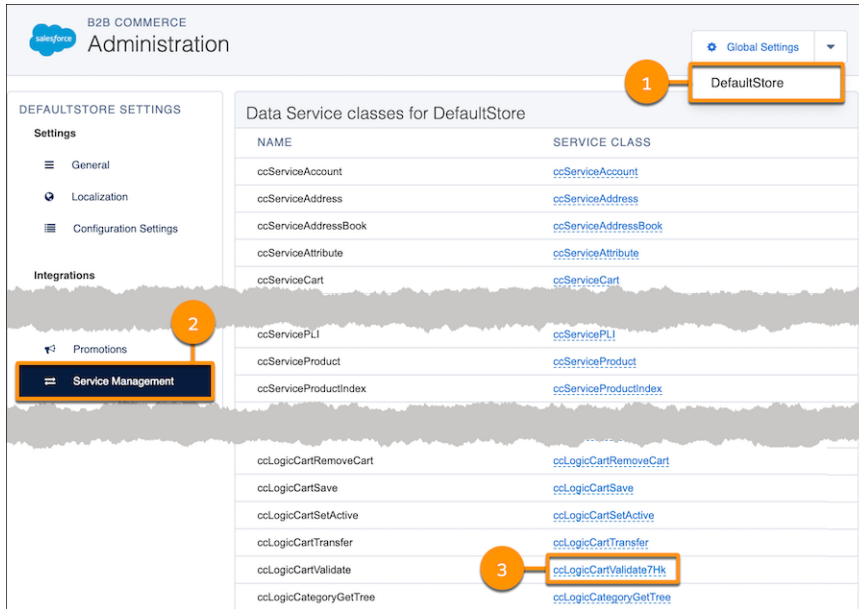
- `ccrz.cc_api_CartExtension.allowCheckout`
- `ccrz.cc_api_CartExtension.getCartMessages`

```
global with sharing class ccrz.ccLogicCartValidate7Hk extends ccrz.ccLogicCartValidate {
    global virtual override Map<String, Object> processValidate(Map<String, Object>
) {
        Map<String, Object> cartFetchResults = (Map<String, Object>)ccUtil.defv(,
ccrz.ccApiCart.RESPONSE, new Map<String, Object> {
            ccrz.ccApi.SUCCESS => false
        });
        try {
            ccrz.cc_api_cartExtension cartExtension =
ccrz.cc_api_cartExtension.getInstance();
            .put(ccrz.ccApiCart.ALLOW_CHECKOUT,
cartExtension.allowCheckout());
            List<ccrz.cc_bean_Message> messages = cartExtension.getCartMessages();

            if (ccrz.ccUtil.isEmpty(messages)) {
                .put(ccrz.ccApi.MESSAGES, messages);
            }
        }
        return ;
    }
}
```

```
}
}
```

To enable this extension, update your storefront settings in CC Admin. Select your storefront (1), and then select **Service Management** in the Integrations section of the Storefront Settings menu (2). Update the `ccLogicCartValidate` class (3) with `ccLogicCartValidate7Hk`.



**Tip:** For more help with extending a default logic service provider, including considerations and conventions, see [Extend or Override a Default Logic Service Provider](#).

## ccrz.ccLogicCategoryGetTree

Define the business logic for returning a generated category tree from a public cache.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiCategory.getTree](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

#### [processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

#### [getTree](#)

Calls the `ccrz.ccApiPublicCache.fetch` method for returning the cached category tree data.

#### [prepReturn](#)

Provides a placeholder for overriding the base `ccrz.ccLogic.prepReturn` method, but doesn't execute any default logic.

#### [ccrz.ccLogicCategoryGetTreeHk Example Extension Class](#)

The managed package provides an example extension class, `ccrz.ccLogicCategoryGetTreeHk`, that you can use as a starting point for extending `ccrz.ccLogicCategoryGetTree` to reference legacy methods of the `ccrz.cc_hk_Category` extension point class.

## ccrz.ccLogicCategoryGetTree.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCategory.getTree` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCategoryGetTree` class in the following order:

1. `processInput`
2. `getTree`
3. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCategoryGetTree.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)


This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCategoryGetTree.process` method.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

**ccrz.ccApiCategory.CATEGORYLOCALE**

String that specifies a supported locale to return the category tree for.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccApiCategory.STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

## ccrz.ccLogicCategoryGetTree.getTree

Calls the `ccrz.ccApiPublicCache.fetch` method for returning the cached category tree data.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> getTree (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working Map<String, Object> after the `ccrz.ccLogicCategoryGetTree.processInput` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiCategory.CATEGORYLOCALE**

String that specifies a supported locale to return the category tree for.

**ccrz.ccApiCategory.STOREFRONT**

String that specifies the name of a storefront.

### Logic Performed

This method completes the following actions:

1. Invokes the `ccrz.ccLogic.pullFromCache` public utility method, which calls `ccrz.ccLogic.prepareCacheKey`.
2. Passes input data to the `ccrz.ccApiPublicCache.fetch` method:

Logic Chain Input	ccrz.ccApiSiteIndex.fetch Input
<code>ccrz.ccCategoryCacheBuilder.CACHE_NAME</code>	<code>ccrz.ccApiPublicCache.CACHE_NAME</code>
<code>ccrz.ccApiCategory.CATEGORYLOCALE</code>	<code>ccrz.ccApiPublicCache.LOCALE</code>
<code>ccrz.ccApiCategory.STOREFRONT</code>	<code>ccrz.ccApiPublicCache.STORE_NAME</code>

3. Deserializes the `ccrz.ccApiPublicCache.CONTENT` output from `ccrz.ccApiPublicCache.fetch`.
4. Calls a public override of the `ccrz.ccLogic.prepareCacheData` method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiCategory.CATEGORYTREE`

`List<Object>` of the cached category tree data, deserialized from the `ccrz.ccApiPublicCache.CONTENT` output of `ccrz.ccApiPublicCache.fetch`.

### Example: Example

Override the default `ccrz.ccLogicCategoryGetTree.getTree` method to rename the top-level categories.

```
global virtual with sharing class renameTopLevelCategories extends
ccrz.ccLogicCategoryGetTree {
    global virtual override Map<String, Object> getTree(Map<String, Object> inputData)
    {
        Map<String, Object> getTreeResults = super.getTree(inputData);

        List<Object> categoryTree =
(List<Object>) defaultGetTreeResults.get(ccrz.ccApiCategory.CATEGORYTREE);

        String storefront = (String) inputData.get(ccrz.ccApiCategory.STOREFRONT);

        for(Object category:categoryTree) {
            Map<String, Object> categoryMap = (Map<String, Object>) category;
            categoryMap.put('name', storefront + '-' + categoryMap.get('name'));
        }

        return getTreeResults;
    }
}
```

SEE ALSO:

[ccrz.ccApiPublicCache.fetch](#)

## ccrz.ccLogicCategoryGetTree.prepReturn

Provides a placeholder for overriding the base `ccrz.ccLogic.prepReturn` method, but doesn't execute any default logic.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCategoryGetTree.getTree` method completes.

By default, this method doesn't evaluate any input data.

### Outputs

This method returns the working `Map<String, Object>`, but by default doesn't set any key values.

## ccrz.ccLogicCategoryGetTreeHk Example Extension Class

The managed package provides an example extension class, `ccrz.ccLogicCategoryGetTreeHk`, that you can use as a starting point for extending `ccrz.ccLogicCategoryGetTree` to reference legacy methods of the `ccrz.cc_hk_Category` extension point class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Important:** This example extension class is intended for short-term compatibility with existing extensions of `ccrz.cc_hk_Category`, which is deprecated. For new solutions, extend `ccrz.ccLogicCategoryGetTree`.

This extension overrides the default `ccrz.ccLogicCategoryGetTree.getTree` method to call the `ccrz.cc_hk_Category.fetch` method.

To enable this extension, update your storefront settings in CC Admin. Select your storefront (1), and then select **Service Management** in the Integrations section of the Storefront Settings menu (2). Update the `ccLogicCategoryGetTree` class (3) with `ccLogicCategoryGetTreeHk`.

The screenshot shows the B2B Commerce Administration interface. On the left, the 'DEFAULTSTORE SETTINGS' menu is visible, with 'Integrations' expanded and 'Service Management' selected (marked with a '2'). The main content area shows 'Data Service classes for DefaultStore' (marked with a '1'). A table lists various service classes and their corresponding implementations. The row for 'ccLogicCategoryGetTree' is highlighted, with its implementation 'ccLogicCategoryGetTreeHk' selected (marked with a '3').

NAME	SERVICE CLASS
ccServiceAccount	ccServiceAccount
ccServiceAddress	ccServiceAddress
ccServiceAddressBook	ccServiceAddressBook
ccServiceAttribute	ccServiceAttribute
ccServiceCart	ccServiceCart
ccServicePLI	ccServicePLI
ccServiceProduct	ccServiceProduct
ccServiceProductIndex	ccServiceProductIndex
ccLogicOrderAuthCancel	ccLogicOrderAuthCancel
ccLogicCartSetActive	ccLogicCartSetActive
ccLogicCartTransfer	ccLogicCartTransfer
ccLogicCartValidate	ccLogicCartValidate
ccLogicCategoryGetTree	ccLogicCategoryGetTreeHk
ccLogicCategoryTree	ccLogicCategoryTree

**Tip:** For more help with extending a default logic service provider, including considerations and conventions, see [Extend or Override a Default Logic Service Provider](#).

SEE ALSO:

[ccrz.cc\\_hk\\_Category \(Deprecated\)](#)

## ccrz.ccLogicCategoryUrlFor

Define the business logic for returning the friendly URLs for categories.

### Compatibility

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Global API Caller

[ccrz.ccApiCategory.getUrlFor](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

#### [processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

#### [fetchSiteIndices](#)

Calls the `ccrz.ccApiSiteIndex.fetch` method for returning `ccrz__E_SiteIndex__c` records that correspond to the specified categories.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### SEE ALSO:

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

## ccrz.ccLogicCategoryUrlFor.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCategory.getUrlFor` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCategoryUrlFor` class in the following order:

1. `processInput`
2. `fetchSiteIndices`
3. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.



**Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCategoryUrlFor.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCategoryUrlFor.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

**ccrz.ccApiCategory.CATEGORYIDLIST**

List<String> of Salesforce IDs for categories whose URLs you want to return.

## Outputs

This method returns a Map<String, Object>, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

## ccrz.ccLogicCategoryUrlFor.fetchSiteIndices

Calls the `ccrz.ccApiSiteIndex.fetch` method for returning `ccrz__E_SiteIndex__c` records that correspond to the specified categories.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> fetchSiteIndices (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working Map<String, Object> after the `ccrz.ccLogicCategoryUrlFor.processInput` method completes.

This method requires the following key in the working map:

**ccrz.ccApiCategory.CATEGORYIDLIST**

List<String> of Salesforce IDs for categories whose URLs you want to return.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiCategory.CATEGORYLOCALE**

String that specifies one of your storefront's supported locales.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

**ccrz.ccApiCategory.CATEGORYLOCALES**

Set<String> of your storefront's supported locales. The logic chain returns internationalized friendly URL values for each locale.

## Logic Performed

This method completes the following actions:

1. Passes input data to the `ccrz.ccApiSiteIndex.fetch` method:

Logic Chain Input	ccrz.ccApiSiteIndex.fetch Input
<code>ccrz.ccApiCategory.CATEGORYIDLIST</code>	<code>ccrz.ccApiSiteIndex.IDS</code>
<code>ccrz.ccApiCategory.CATEGORYLOCALE</code>	<code>ccrz.ccApiSiteIndex.LOCALE</code>
<code>ccrz.ccApiCategory.CATEGORYLOCALES</code>	<code>ccrz.ccApiSiteIndex.LOCALES</code>

2. Parses the `ccrz.ccApiSiteIndex.SITE_INDICES` output from `ccrz.ccApiSiteIndex.fetch`. For each queried `ccrz__E_SiteIndex__c` record, this method completes the following actions:
  - If the input data specified only a single locale, this method:
    - a. Adds the referenced category ID and friendly URL to `ccrz.ccApiCategory.CATEGORYURLMAP`.
    - b. Adds the referenced category ID and the `ccrz__E_SiteIndex__c` record to `ccrz.ccApiCategory.SITEINDICES`.
  - If the input data specified multiple locales, this method:
    - a. Adds the referenced category ID and locale-specific friendly URL to an inner map, and then adds the inner map to `ccrz.ccApiCategory.LOCALECATEGORYURLMAP`, keyed by locale.
    - b. Adds the referenced category ID and locale-specific `ccrz__E_SiteIndex__c` record to an inner map, and then adds the inner map to `ccrz.ccApiCategory.LOCALESITEINDICES`, keyed by locale.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiCategory.CATEGORYURLMAP**

`Map<String, Object>` for a single locale, where each key is a category ID and each value is the category's friendly URL.

### **ccrz.ccApiCategory.LOCALECATEGORYURLMAP**

`Map<String, Map<String, Object>>` for multiple locales, where:

- The outer map is keyed by category ID.
- Each inner map is keyed by locale.
- Each inner map's value is the category's friendly URL for the locale.

### **ccrz.ccApiCategory.LOCALESITEINDICES**

`Map<String, Map<String, Object>>` for multiple locales, where:

- The outer map is keyed by category ID.
- Each inner map is keyed by locale.
- Each inner map's value is the category's corresponding `ccrz__E_SiteIndex__c` record for the locale.

**ccrz.ccApiCategory.SITEINDICES**

Map<String, Object> for a single locale, where each key is a category ID and each value is the corresponding `ccrz__E_SiteIndex__c` record.

SEE ALSO:

[ccrz\\_\\_E\\_SiteIndex\\_\\_c](#)

[ccrz.ccApiSiteIndex.fetch](#)

[ccrz.ccLogicProductUrlFor.fetchCategoryURLs](#)

**ccrz.ccLogicCategoryUrlFor.prepReturn**

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

**Signature**

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working map `Map<String, Object>` after the `ccrz.ccLogicCategoryUrlFor.fetchSiteIndices` method completes.

**Outputs**

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`


**ccrz.ccLogicCouponApply**

Define the business logic for adding and applying a coupon to a specific shopping cart.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

 **Important:** The methods of this class don't support [Apex callouts](#).

## Global API Caller

[ccrz.ccApiCoupon.apply](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [retrieveCartData](#)

Calls the `ccrz.ccApiCart.fetch` method for returning the cart data to evaluate a coupon against. If the input data already specified a cart and cart items to evaluate, the logic chain skips this method.

#### [addCoupon](#)

Attempts to add a coupon code to the cart, and executes the DML `insert` operation on the corresponding `ccrz__E_CartCoupon__c` record.

#### [applyCoupons](#)

Evaluates the cart items and attempts to apply the coupon's specified discount. This method doesn't execute any DML `upsert` statements, but can remove expired coupons and update or insert cart items and coupons.

#### [performDML](#)

Executes the DML `upsert` operation on cart items and `update` operation on the cart after the coupon's discount is applied.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

#### [handleCouponApplyException](#)

Catches a `ccrz.ccApiCouponApply.COUPONAPPLYEXCEPTION` that occurs during the logic chain, and logs the exception message.

#### [Extension Examples for ccrz.ccLogicCouponApply](#)

To implement custom business logic for coupons, create custom subscriber code that extends the default `ccrz.ccLogicCouponApply` class. The following code examples provide starting points for working around limits in the default coupon functionality and defining custom requirements.

## ccrz.ccLogicCouponApply.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCoupon.apply` method.

### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCouponApply` class in the following order:

1. `processInput`
2. `retrieveCartData`
3. `addCoupon`
4. `applyCoupons`
5. `performDML`
6. `prepReturn`

### Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method catches any exceptions that occur as the method executes.

## ccrz.ccLogicCouponApply.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCouponApply.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

### `ccrz.ccApiCart.CART_ID`

String that specifies the Salesforce ID of the cart that you want to add or apply a coupon to.

### `ccrz.ccApiCart.CART_ENCID`

String that specifies the encrypted ID of the cart that you want to add or apply a coupon to.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### `ccrz.ccApiCoupon.CART_HEADER`

`ccrz__E_Cart__c` record that you want to add or apply a coupon to.

### `ccrz.ccApiCoupon.CART_ITEMS`

List<`ccrz__E_CartItem__c`> of the cart items for the `ccrz.ccApiCoupon.CART_HEADER`, if specified.



**Tip:** If you specify either `ccrz.ccApiCoupon.CART_HEADER` or `ccrz.ccApiCoupon.CART_ITEMS`, make sure to include the other key, as well. When these keys are specified, the logic chain already has the required cart data, and can skip the cart fetch call.

## Logic Performed

This method evaluates whether the input map specifies `ccrz.ccApiCoupon.CART_HEADER` and `ccrz.ccApiCoupon.CART_ITEMS`. If the input map specifies these keys, this method sets the `COUPONSKIPCARTFETCH` output to `true`. This value indicates that the logic chain can skip `ccrz.ccLogicCouponApply.retrieveCartData` because the cart data is already available.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:



**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart specified in `ccrz.ccApiCoupon.CART_HEADER`.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart specified in `ccrz.ccApiCoupon.CART_HEADER`.

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

**COUPONSKIPCARTFETCH**

Boolean

Value	Usage
true	The original input data included <code>ccrz.ccApiCoupon.CART_HEADER</code> and <code>ccrz.ccApiCoupon.CART_ITEMS</code> . Skip the <code>ccrz.ccApiCart.fetch</code> call in <code>ccrz.ccLogicCouponApply.retrieveCartData</code> .
false (default)	The original input data omitted <code>ccrz.ccApiCoupon.CART_HEADER</code> and <code>ccrz.ccApiCoupon.CART_ITEMS</code> . Execute the <code>ccrz.ccApiCart.fetch</code> call in <code>ccrz.ccLogicCouponApply.retrieveCartData</code> for fetching the cart data to evaluate the coupon against.

The `ccrz.ccLogicCouponApply.prepReturn` method removes these intermediate keys from the working map.

## ccrz.ccLogicCouponApply.retrieveCartData

Calls the `ccrz.ccApiCart.fetch` method for returning the cart data to evaluate a coupon against. If the input data already specified a cart and cart items to evaluate, the logic chain skips this method.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global virtual Map<String, Object> retrieveCartData (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working Map<String, Object> from the `ccrz.ccLogicCouponApply.processInput` method.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart that you want to add or apply a coupon to.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart that you want to add or apply a coupon to.

**COUPONSKIPCARTFETCH**

Boolean

Value	Usage
<i>true</i>	The original input data included <code>ccrz.ccApiCoupon.CART_HEADER</code> and <code>ccrz.ccApiCoupon.CART_ITEMS</code> . This method is skipped.
<i>false</i> (default)	The input data omitted <code>ccrz.ccApiCoupon.CART_HEADER</code> and <code>ccrz.ccApiCoupon.CART_ITEMS</code> . This method executes.

## Logic Performed

This method completes the following actions:

1. Evaluates the value of the `COUPONSKIPCARTFETCH` input.
2. Passes `ccrz.ccApiCart.CART_ID` or `ccrz.ccApiCart.CART_ENCID` as input for [ccrz.ccApiCart.fetch](#).
3. Parses the `ccrz.ccApiCart.CART_OBJLIST` that `ccrz.ccApiCart.fetch` returns.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the fetched cart.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the fetched cart.

**ccrz.ccApiCoupon.CART\_HEADER**

`ccrz__E_Cart__c` record specified in the original input data or that the `ccrz.ccApiCart.fetch` method returned.

**ccrz.ccApiCoupon.CART\_ITEMS**

`List<ccrz__E_CartItem__c>` of the cart items for the `ccrz.ccApiCoupon.CART_HEADER` specified in the original input data or that the `ccrz.ccApiCart.fetch` method returned.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicCouponApply.addCoupon

Attempts to add a coupon code to the cart, and executes the DML `insert` operation on the corresponding `ccrz__E_CartCoupon__c` record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> addCoupon (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCouponApply.retrieveCartData` method.

This method requires the following keys in the working map:

### `ccrz.ccApiCoupon.CART_HEADER`

`ccrz__E_Cart__c` record to add a coupon to.

### `ccrz.ccApiCoupon.CART_ITEMS`

`List<ccrz__E_CartItem__c>` of the cart items for the `ccrz.ccApiCoupon.CART_HEADER`.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiCoupon.CODE`

String that specifies a coupon code to add to the cart.

## Logic Performed

This method passes `ccrz.ccApiCoupon.CART_HEADER`, `ccrz.ccApiCoupon.CART_ITEMS`, and `ccrz.ccApiCoupon.CODE` as input for the `ccrz.cc_api_Cart.addCoupon` public method, which completes the following actions:

1. Evaluates the coupon code.
2. Creates a `ccrz__E_CartCoupon__c` record.
3. Executes the DML `insert` operation for the `ccrz__E_CartCoupon__c`.

## Outputs


This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApi.MESSAGES`

`List<ccrz.cc_bean_Message>`, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is `FALSE`.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

SEE ALSO:

[ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)[Inserting and Updating Records](#)

## ccrz.ccLogicCouponApply.applyCoupons

Evaluates the cart items and attempts to apply the coupon's specified discount. This method doesn't execute any DML `upsert` statements, but can remove expired coupons and update or insert cart items and coupons.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global virtual Map<String, Object> applyCoupons (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCouponApply.addCoupon` method.

This method requires the following keys in the working map:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart that you want to apply the coupon to.

**ccrz.ccApiCoupon.CART\_ITEMS**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> of the cart items for the `ccrz.ccApiCoupon.CART_HEADER`.

**ccrz.ccApiCoupon.SKIP\_APPLY\_CALC**

Boolean

Value	Usage
<i>true</i>	Skip this method's execution, and don't apply the coupon's discount.
<i>false</i> (default)	Execute this method.

**Logic Performed**

This method executes only when `ccrz.ccApiCoupon.SKIP_APPLY_CALC` is `false`. This method passes `ccrz.ccApiCart.CART_ID` and `ccrz.ccApiCoupon.CART_ITEMS` as input for the `ccrz.cc_util_Coupon.applyCouponsToCart` public method, which completes the following actions:

1. Parses `ccrz.ccApiCoupon.CART_ITEMS` and identifies the major line items.
2. Queries for `ccrz__E_CartCoupon__c` records already related to the cart.
3. Validates each `ccrz__E_CartCoupon__c` record for start and end dates, and deletes any invalid coupons.
4. Updates cart items.
5. Evaluates the target and source criteria for the coupon, and applies the coupon's discount when the criteria are satisfied.

**Outputs**

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)[Inserting and Updating Records](#)[Deleting Records](#)**ccrz.ccLogicCouponApply.performDML**

Executes the DML `upsert` operation on cart items and `update` operation on the cart after the coupon's discount is applied.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> performDML (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCouponApply.applyCoupons` method.

This method requires the following keys in the working map:

**ccrz.ccApiCoupon.CART\_HEADER**

`ccrz__E_Cart__c` record to update.

**ccrz.ccApiCoupon.CART\_ITEMS**

`List<ccrz__E_CartItem__c>` of the cart items to upsert.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)

[Inserting and Updating Records](#)

[Upserting Records](#)

## ccrz.ccLogicCouponApply.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCouponApply.performDML` method.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApiCoupon.CART_HEADER`
- `ccrz.ccApiCoupon.CART_ITEMS`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `COUPONSKIPCARTFETCH`

## ccrz.ccLogicCouponApply.handleCouponApplyException

Catches a `ccrz.ccApiCouponApply.COUPONAPPLYEXCEPTION` that occurs during the logic chain, and logs the exception message.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> handleCouponApplyException (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that can include the following keys:

**ccrz.ccLogicCouponApply.COUPONAPPLYEXCEPTION**  
 CouponApplyException that occurred during the process.

## Outputs

None

## Extension Examples for ccrz.ccLogicCouponApply

To implement custom business logic for coupons, create custom subscriber code that extends the default `ccrz.ccLogicCouponApply` class. The following code examples provide starting points for working around limits in the default coupon functionality and defining custom requirements.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Important:

- The following examples are for demo purposes only, and don't provide a complete extension solution. Often, extension is a more complex customization that includes object data, data service provider, logic service provider, and buyer experience extensions.
- For more information about writing a logic service provider extension and configuring your storefront to reference your extension, see [Extend or Override a Default Logic Service Provider](#).

### Example: **Example: Restrict a Coupon to a Specific Account**

The following `ccrz.ccLogicCouponApply` extension implements the following custom logic:

- Defines a custom exception class, `customCouponApplyException`, for gracefully exiting the process.
- Overrides the default `ccrz.ccLogicCouponApply.process` method to add custom error handling.
- Overrides the default `ccrz.ccLogicCouponApply.addCoupon` method to evaluate a custom field on the coupon code's corresponding `ccrz__E_Coupon__c` record. The custom field, `ccrz__EligibleAccounts__c`, is a list of Salesforce IDs for accounts that are eligible to use the coupon.
  - If the current account is included in the list of eligible accounts, the `ccrz.ccLogicCouponApply.addCoupon` method completes normally.



**Note:** The `ccrz.cc_CallContext.currAccount.Id` value specifies the account's 18-digit ID.

- If the current account is excluded from the list of eligible accounts, the method returns error information as a `ccrz.cc_bean_Message` and throws the custom exception class, `customCouponApplyException`.

```
global with sharing class customCouponApplyLogic {
    global class customCouponApplyException extends Exception{}

    global with sharing class restrictCouponAccount extends ccrz.ccLogicCouponApply {

        global override Map<String, Object> process(Map<String, Object> inputData) {
            try {
                inputData = super.process(inputData);
            } catch (customCouponApplyLogic.customCouponApplyException e) {
                // Log and handle the exception
            }
            return super.prepReturn(inputData);
        }

        global override Map<String, Object> addCoupon(Map<String, Object> inputData)
    {
        ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:E',
'restrictCouponAccount.addCoupon');
        ccrz.E_Coupon__c couponToEvaluate = // Evaluate the ccrz.ccApiCoupon.CODE
```



```

input and select its coupon record
    Set<String> eligibleAccounts = couponToEvaluate.EligibleAccounts__c;

    if(eligibleAccounts == null ||
eligibleAccounts.contains(ccrz.cc_CallContext.currAccount.Id)) {
        super.addCoupon(inputdata);
    } else {
        ccrz.cc_bean_Message msg = new ccrz.cc_bean_Message();
        msg.labelId = 'CouponAddMsg_InvalidCoupon';
        msg.classToAppend = 'couponMessagingSection-Error';
        msg.type = ccrz.cc_bean_Message.MessageType.CUSTOM;
        msg.severity = ccrz.cc_bean_Message.MessageSeverity.ERROR;
        inputdata.put(ccrz.ccApi.MESSAGES, new List<ccrz.cc_bean_Message>{
            msg
        });
        inputdata.put(ccrz.ccApi.SUCCESS, false);
        ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X',
'restrictCouponAccount.addCoupon.exception');
        throw new customCouponApplyLogic.customCouponApplyException('This
account isn not eligible for coupons');
    }
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X',
'restrictCouponAccount.addCoupon');
    return inputdata;
}
}
}

```

### Example: Example: Automatically Add a Coupon for Certain Accounts

The following `ccrz.ccLogicCouponApply` extension implements the following custom logic:

1. Defines a custom exception class, `customCouponApplyException`, for gracefully exiting the process.
2. Overrides the default `ccrz.ccLogicCouponApply.addCoupon` method to:
  - a. Build the list of accounts that are eligible for automatic coupons
  - b. Evaluate whether the current account is included in the list of eligible accounts.
  - c. If the current account is eligible, determine whether the cart already has a coupon applied.
  - d. If the cart doesn't have a coupon already, automatically apply a coupon and let the `ccrz.ccLogicCouponApply.addCoupon` method completes normally.

### Tip:

- Consider modifying this example so that the method queries coupons that are accounts are eligible for and evaluate other requirements. This example shows how to apply a specific code to an eligible account.
- This example functions correctly only when the [Optimize Cart Repricing configuration of the Cart module](#) is a value *other than* Always. For other values, the `ccrz.ccLogicCartPrice.applyCoupons` method invokes `ccrz.ccLogicCouponApply` logic normally.

```

global with sharing class customCouponApplyLogic {
    global class customCouponApplyException extends Exception{}
}

```

```

    global with sharing class autoAddCouponForCertainAccounts extends
ccrz.ccLogicCouponApply {
    global override Map<String, Object> addCoupon(Map<String, Object> inputdata)
    {
        ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:E',
'autoAddCouponForCertainAccounts.addCoupon');

        // Add a SQL query for coupons that can be applied for these accounts
        // Accounts that are eligible for automatically applying a coupon
        Set<String> eligibleAccounts = new Set<String>{account_ID_1, account_ID_2};

        if(eligibleAccounts.contains(ccrz.cc_CallContext.currAccount.Id)) {
            String cartId = (String) inputdata.get(ccrz.ccApiCart.CART_ID);
            List<ccrz__E_CartCoupon__c> couponsAppliedToCart = [SELECT ID FROM
ccrz__E_CartCoupon__c WHERE ccrz__Cart__r.Id = :cartId];
            if(ccrz.ccUtil.isEmpty(couponsAppliedToCart) &&
!ccrz.ccUtil.isKeyValued(inputdata, ccrz.ccApiCoupon.CODE)) {
                // Apply a hard-coded coupon, or parse a queried coupon for its
code
                inputdata.put(ccrz.ccApiCoupon.CODE, 'couponCode');
            }
        }
        super.addCoupon(inputdata);
        ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X',
'autoAddCouponForCertainAccounts.addCoupon');
        return inputdata;
    }
}

```



### Example: Example: Apply a Discount to an Attribute-Driven Commerce Product

By default, only a cart total coupon is compatible with attribute-driven commerce (ADC). The following `ccrz.ccLogicCouponApply` extension implements a \$0 absolute cart total discount (for coupon tracking and reporting) and a \$10 absolute discount on individual cart items. The discount applies only to cart items that match certain combinations of attributes. This extension implements the following custom logic:

1. Defines a custom exception class, `customCouponApplyException`, for gracefully exiting the process.
2. Overrides the default `ccrz.ccLogicCouponApply.process` method to add custom error handling.
3. Overrides the default `ccrz.ccLogicCouponApply.addCoupon` method to:
  - a. Query the ADC products that are eligible for the coupon. One option for determining eligibility is to use custom fields on the `ccrz__E_Coupon__c` object, which reference a specific combination of eligible primary and secondary attributes.
  - b. Parse the cart input data, and evaluate whether any cart items match the eligible attributes. Add the cart items to a list to pass to the `ccrz.ccLogicCouponApply.applyCoupons` method.
    - If a cart item matches the eligible attributes, add the cart item to a list, and let the `ccrz.ccLogicCouponApply.addCoupon` method execute normally.
    - If no cart item matches the eligible attributes, the method returns error information as a `ccrz.cc_bean_Message` and throws the custom exception class, `customCouponApplyException`.

4. Overrides the default `ccrz.ccLogicCouponApply.applyCoupons` method to:
  - a. Let the `ccrz.ccLogicCouponApply.applyCoupons` method execute normally, which applies a \$0 cart total discount.
  - b. Loop through the cart items that match the eligible attributes, and individually implement a \$10 absolute discount by explicitly setting the `ccrz__AbsoluteDiscount__c` and `ccrz__AdjustmentAmount__c` fields.

**!** **Important:** This example doesn't show how to remove coupon discounts that affect minor line items, which B2B Commerce for Visualforce doesn't support by default. This example also doesn't show how to update the Handlebars templates that show discount information on the storefront. To fully develop and test this customization in your storefront, further extension is necessary.

```
global with sharing class customCouponApplyLogic {
    global class customCouponApplyException extends Exception{}

    global with sharing class customCoupon10OffADCProduct extends ccrz.ccLogicCouponApply
    {
        global override Map<String, Object> process(Map<String, Object> inputData) {
            try {
                inputData = super.process(inputData);
            } catch (customCouponApplyLogic.customCouponApplyException e) {
                // Log and handle the exception
            }
            return super.prepReturn(inputData);
        }

        global override Map<String, Object> addCoupon(Map<String, Object> inputData)
        {
            ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:E',
'customCoupon10OffADCProduct.addCoupon');

            // Fetch a map of products that eligible for the coupon
            Map<String, Object> eligibleADCProducts = new Map<String, Object> {
                eligibleADCProductIDs = New List<String>
{eligibleSelectedPrimaryAttributeID, eligibleSelectedSecondaryAttributeID}
            }

            List<String> cartItemsToDiscount = new List<String>();

            // Find cart items that match eligible attributes
            List<ccrz__E_CartItem__c> cartItems = (List<ccrz__E_CartItem__c>)
inputData.get(ccrz.ccApiCoupon.CART_ITEMS);
            ccrz.ccLog.log(System.LoggingLevel.INFO, 'cartItems', cartItems);
            for (ccrz__E_CartItem__c cartItem : cartItems) {
                if (String.isEmpty(cartItem.ccrz__PrimaryAttr__c)) {
                    List<String> eligibleAttributeCombinations = (List<String>)
eligibleADCProducts.get(cartItem.ccrz__Product__c);
                    if (ccrz.ccUtil.isEmpty(eligibleAttributeCombinations)) {
                        for (String combination : eligibleAttributeCombinations) {
                            if (combination.contains(cartItem.ccrz__PrimaryAttr__c))
{
                                if
(combination.contains(cartItem.ccrz__SecondaryAttr__c)) {
```

```

                cartItemsToDiscount.add(cartItem.Id);
            }
        }
    }
}

// Add the cart total coupon to the cart
if (ccrz.ccUtil.isNotEmpty(cartItemsToDiscount)) {
    inputdata.put('cartItemsToDiscount', cartItemsToDiscount);
    super.addCoupon(inputdata);
} else {
    ccrz.cc_bean_Message msg = new ccrz.cc_bean_Message();
    msg.labelId = 'CouponAddMsg_InvalidCoupon';
    msg.classToAppend = 'couponMessagingSection-Error';
    msg.type = ccrz.cc_bean_Message.MessageType.CUSTOM;
    msg.severity = ccrz.cc_bean_Message.MessageSeverity.ERROR;
    inputdata.put(ccrz.ccApi.MESSAGES, new List<ccrz.cc_bean_Message>{
        msg
    });
    inputdata.put(ccrz.ccApi.SUCCESS, false);
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X',
'customCoupon10OffADCProduct.addCoupon.exception');
    throw new customCouponApplyLogic.customCouponApplyException('This
account isn not eligible for coupons');
}
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X',
'customCoupon10OffADCProduct.addCoupon');
    return inputdata;
}

    global override Map<String, Object> applyCoupons (Map<String, Object> inputdata)
    {
        ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:E',
'customCoupon10OffADCProduct.applyCoupons');

        super.applyCoupons (inputdata);

        // Apply the coupon's discount to the eligible cart items
        List<String> cartItemsToDiscount = (List<String>)
inputdata.get('cartItemsToDiscount');
        if (ccrz.ccUtil.isNotEmpty(cartItemsToDiscount)) {
            List<ccrz__E_CartItem__c> cartItems = (List<ccrz__E_CartItem__c>)
inputdata.get(ccrz.ccApiCoupon.CART_ITEMS);
            for (ccrz__E_CartItem__c cartItem : cartItems) {
                if (cartItemsToDiscount.contains(cartItem.Id)) {
                    if (cartItem.ccrz__Quantity__c != 0 && cartItem.ccrz__Price__c
!= 0) {
                        cartItem.ccrz__AbsoluteDiscount__c = (cartItem.ccrz__Price__c
* .1).setScale(2, System.RoundingMode.HALF_EVEN) * cartItem.ccrz__Quantity__c;
                        cartItem.ccrz__AdjustmentAmount__c =
cartItem.ccrz__AbsoluteDiscount__c * -1;

```

```

        }
    }
}

ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X',
'customCoupon10OffADCProduct.applyCoupons');
return inputData;
}
}
}

```

**SEE ALSO:**

[Offer Coupons for Discounts on Storefront Purchases](#)

## ccrz.ccLogicCouponRemove

Define the business logic for removing coupons that are already applied to a specified shopping cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Global API Caller

[ccrz.ccApiCoupon.remove](#)

### Methods

**IN THIS SECTION:**[process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

[processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

[fetchCartData](#)

Calls the `ccrz.ccApiCart.fetch` method for returning the cart and cart items to remove coupons from.

### remove

Queries the `ccrz__E_CartCoupon__c` records to delete, and resets the discount amounts applied to the `ccrz__E_CartItem__c` records.

### performDML

Executes the DML `delete` operation on the `ccrz__E_CartCoupon__c` records and the `update` operation on the `ccrz__E_Cart__c` and `ccrz__E_CartItem__c` records.

### prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicCouponRemove.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCoupon.remove` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicCouponRemove` class in the following order:

1. `processInput`
2. `fetchCartData`
3. `remove`
4. `performDML`
5. `prepReturn`

### Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCouponRemove.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCouponRemove.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to remove coupons from.

**ccrz.ccApiCart.CART\_ENCID**


String that specifies the encrypted ID of the cart to remove coupons from.

### Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The [ccrz.ccLogicCouponRemove.prepReturn](#) method removes these intermediate keys from the working map.

**ccrz.ccLogicCouponRemove.fetchCartData**

Calls the [ccrz.ccApiCart.fetch](#) method for returning the cart and cart items to remove coupons from.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

**Signature**

```
global virtual Map<String, Object> fetchCartData (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> from the [ccrz.ccLogicCouponRemove.processInput](#) method.

This method can accept either of the following parameters:

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to remove coupons from.

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to remove coupons from.

**Logic Performed**

This method completes the following actions:

1. Passes [ccrz.ccApiCart.CART\\_ID](#) or [ccrz.ccApiCart.CART\\_ENCID](#) as input for [ccrz.ccApiCart.fetch](#).
2. Parses the [ccrz.ccApiCart.CART\\_OBJLIST](#) that [ccrz.ccApiCart.fetch](#) returns.

**Outputs**

This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccLogicCouponRemove.CART\_DATA**

[ccrz\\_\\_E\\_Cart\\_\\_c](#) record that you want to remove coupons from.



**ccrz.ccLogicCouponRemove.CART\_ITEMS\_TO\_UPDATE**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> of the cart items for the `ccrz.ccLogicCouponRemove.CART_DATA`.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

**ccrz.ccLogicCouponRemove.remove**

Queries the `ccrz__E_CartCoupon__c` records to delete, and resets the discount amounts applied to the `ccrz__E_CartItem__c` records.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

**Signature**

```
global virtual Map<String, Object> remove (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCouponRemove.fetchCartData` method.

This method requires the following keys in the working map:

**ccrz.ccLogicCouponRemove.CART\_DATA**

`ccrz__E_Cart__c` record that you want to remove coupons from.

**ccrz.ccLogicCouponRemove.CART\_ITEMS\_TO\_UPDATE**

List<`ccrz__E_CartItem__c`> of the cart items for the `ccrz.ccLogicCouponRemove.CART_DATA`.

**Logic Performed**

This method completes the following actions:

1. Evaluates `ccrz.ccLogicCouponRemove.CART_DATA` and finds the `ccrz__E_CartCoupon__c` records associated with the cart.
2. Evaluates `ccrz.ccLogicCouponRemove.CART_ITEMS_TO_UPDATE` and updates the following fields on each `ccrz__E_CartItem__c`:

Field	Value
ccrz__AbsoluteDiscount__c	<i>null</i>
ccrz__Coupon__c	<i>null</i>
ccrz__PercentDiscount__c	<i>null</i>

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the cart to remove coupons from.

### **ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the cart to remove coupons from.

### **ccrz.ccLogicCouponRemove.CART\_COUPONS\_FOR\_DELETION**

List<[ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)> of the cart coupons to remove.

### **ccrz.ccLogicCouponRemove.CART\_ITEMS\_TO\_UPDATE**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> of the cart items with reset discount amounts.

SEE ALSO:

[ccrz\\_\\_E\\_CartCoupon\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.ccLogicCouponRemove.performDML

Executes the DML `delete` operation on the [ccrz\\_\\_E\\_CartCoupon\\_\\_c](#) records and the `update` operation on the [ccrz\\_\\_E\\_Cart\\_\\_c](#) and [ccrz\\_\\_E\\_CartItem\\_\\_c](#) records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> performDML (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCouponRemove.remove` method.

This method can accept either of the following parameters:

**ccrz.ccLogicCouponRemove.CART\_COUPONS\_FOR\_DELETION**

`List<ccrz__E_CartCoupon__c>` of the cart coupons to remove.

**ccrz.ccLogicCouponRemove.CART\_DATA**

`ccrz__E_Cart__c` record that you want to remove coupons from.

**ccrz.ccLogicCouponRemove.CART\_ITEMS\_TO\_UPDATE**

`List<ccrz__E_CartItem__c>` of the cart items for the `ccrz.ccLogicCouponRemove.CART_DATA`.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccApiCoupon.REPRICE\_CART**

Boolean

Value	Usage
<code>true</code>	Chain a call to the <code>ccrz.ccApiCart.price</code> method to reprice the cart after removing the coupons.  <b>Warning:</b> This chained call isn't safe for Apex callouts.
<code>false</code> (default)	Don't chain a call to the <code>ccrz.ccApiCart.price</code> method.

## Logic Performed

This method completes the following actions:

1. Executes the DML `delete` operation on `ccrz.ccLogicCouponRemove.CART_COUPONS_FOR_DELETION`.
2. Executes the DML `update` operation on `ccrz.ccLogicCouponRemove.CART_ITEMS_TO_UPDATE`.
3. Executes the DML `update` operation on `ccrz.ccLogicCouponRemove.CART_DATA`.
4. Evaluates `ccrz.ccApiCoupon.REPRICE_CART`, and calls `ccrz.ccApiCart.price` to reprice the cart after removing the coupons.


## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.

Value	Usage
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

## SEE ALSO:

[ccrz.ccApiCart.price](#)[Inserting and Updating Records](#)[Upserting Records](#)

## ccrz.ccLogicCouponRemove.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicCouponRemove.performDML` method.

### Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicCouponRemove.CART_COUPONS_FOR_DELETION`
- `ccrz.ccLogicCouponRemove.CART_DATA`
- `ccrz.ccLogicCouponRemove.CART_ITEMS_TO_UPDATE`

## ccrz.ccLogicCouponUpdateUse

Define the business logic for removing coupons that are already applied to a specified shopping cart.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Global API Caller

[ccrz.ccApiCoupon.updateOnUse](#)

### Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

[processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

[fetchCoupon](#)

Calls the `ccrz.ccApiCoupon.fetch` method for returning the coupons to update.

[updateCoupon](#)

Parses the fetched coupon data and updates each coupon's `ccrz__TotalUsed__c` field by the `ccrz.ccApiCoupon.INCREMENT_AMOUNT` value.

[performDML](#)

Executes the DML `update` operation on the `ccrz__E_Coupon__c` records.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### ccrz.ccLogicCouponUpdateUse.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiCoupon.updateOnUse` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicCouponUpdateUse` class in the following order:

1. `processInput`
2. `fetchCoupon`
3. `updateCoupon`
4. `performDML`
5. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.



**Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicCouponUpdateUse.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicCouponUpdateUse.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

### `ccrz.ccApiCoupon.CODE`

String that specifies the value of the `ccrz__CouponCode__c` field of the coupon to update.

### `ccrz.ccApiCoupon.CODES`

`List<String>` that specifies `ccrz__CouponCode__c` field values of the coupons to update.

### `ccrz.ccApiCoupon.ID`

String that specifies the Salesforce ID of the coupon to update.

### `ccrz.ccApiCoupon.IDS`

`List<String>` that specifies the Salesforce IDs of coupons to update.

 **Note:** This method checks for these keys in the following order of precedence:

1. `ccrz.ccApiCoupon.ID`
2. `ccrz.ccApiCoupon.IDS`
3. `ccrz.ccApiCoupon.CODE`
4. `ccrz.ccApiCoupon.CODES`

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### `ccrz.ccApiCoupon.INCREMENT_AMOUNT`


A positive or negative integer to add to the `ccrz__TotalUsed__c` field of the specified coupons.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### `ccrz.ccApi.SUCCESS`


Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

### `ccrz.ccApiCoupon.LOCK_COUPON`

Boolean

Value	Usage
<code>true</code> (default)	Query the specified coupons using the SOQL <code>FOR UPDATE</code> keyword, which locks the coupons from other changes until the update completes.
<code>false</code>	Don't lock coupons during this method's execution.

 **Note:** If the original input data specified a value for this input, this method uses that value. Otherwise, this method sets a default value of `true`.

#### **ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The `ccrz.ccLogicCouponUpdateUse.prepReturn` method removes these intermediate keys from the working map.

## ccrz.ccLogicCouponUpdateUse.fetchCoupon

Calls the `ccrz.ccApiCoupon.fetch` method for returning the coupons to update.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Signature

```
global virtual Map<String, Object> fetchCoupon (Map<String, Object>)
```

### Inputs (Required)

This method accepts the Map<String, Object> after the `ccrz.ccLogicCouponUpdateUse.processInput` method completes.

This method also requires one of the following keys in the working map:

#### **ccrz.ccApiCoupon.CODE**

String that specifies the value of the `ccrz__CouponCode__c` field of the coupon to update.

#### **ccrz.ccApiCoupon.CODES**

List<String> that specifies `ccrz__CouponCode__c` field values of the coupons to update.

#### **ccrz.ccApiCoupon.ID**

String that specifies the Salesforce ID of the coupon to update.

#### **ccrz.ccApiCoupon.IDS**

List<String> that specifies the Salesforce IDs of coupons to update.



## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccApiCoupon.LOCK\_COUPON**

Boolean

Value	Usage
<i>true</i> (default)	Query the specified coupons using the SOQL <code>FOR UPDATE</code> keyword, which locks the coupons from other changes until the update completes.
<i>false</i>	Don't lock coupons during this method's execution.

## Logic Performed

This method completes the following actions:

1. Passes `ccrz.ccApiCoupon.ID`, `ccrz.ccApiCoupon.IDS`, `ccrz.ccApiCoupon.CODE`, or `ccrz.ccApiCoupon.CODES` as input for the `ccrz.ccApiCoupon.fetch` method. This method also passes `ccrz.ccApiCoupon.LOCK_COUPON` as the `ccrz.ccService.FORUPDATE` input for `ccrz.ccApiCoupon.fetch`.
2. Stores the `ccrz.ccApiCoupon.fetch` output data in the `ccrz.ccLogicCouponUpdateUse.COUPON_RETURN_DATA` intermediate key.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccLogicCouponUpdateUse.COUPON\_RETURN\_DATA**

`Map<String, Object>` that the `ccrz.ccApiCoupon.fetch` method returned.

SEE ALSO:

[ccrz.ccApiCoupon.fetch](#)

## ccrz.ccLogicCouponUpdateUse.updateCoupon

Parses the fetched coupon data and updates each coupon's `ccrz__TotalUsed__c` field by the `ccrz.ccApiCoupon.INCREMENT_AMOUNT` value.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> updateCoupon (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCouponUpdateUse.fetchCoupon` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiCoupon.INCREMENT\_AMOUNT**

A positive or negative integer to add to the `ccrz__TotalUsed__c` field of the specified coupons.

### **ccrz.ccLogicCouponUpdateUse.COUPON\_RETURN\_DATA**

`Map<String, Object>` that the `ccrz.ccApiCoupon.fetch` method returned during `ccrz.ccLogicCouponUpdateUse.fetchCoupon`.

## Logic Performed

This method completes the following actions for each `ccrz__E_Coupon__c` record included in `ccrz.ccLogicCouponUpdateUse.COUPON_RETURN_DATA`:

1. Updates the value of the `ccrz__TotalUsed__c` field by the `ccrz.ccApiCoupon.INCREMENT_AMOUNT`.
2. Adds the updated coupon to the `ccrz.ccApiCoupon.COUPONLIST` output.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiCoupon.COUPONLIST**

`Map<String, ccrz__E_Coupon__c>`, keyed by Salesforce ID, of the `ccrz__E_Coupon__c` records with incremented `ccrz__TotalUsed__c` values.

SEE ALSO:

[ccrz\\_\\_E\\_Coupon\\_\\_c](#)  
[ccrz.ccApiCoupon.fetch](#)

## ccrz.ccLogicCouponUpdateUse.performDML

Executes the DML `update` operation on the `ccrz__E_Coupon__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> performDML (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicCouponUpdateUse.updateCoupon` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiCoupon.COUPONLIST`

`Map<String, ccrz__E_Coupon__c>`, keyed by Salesforce ID, of the `ccrz__E_Coupon__c` records with incremented `ccrz__TotalUsed__c` values.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiCoupon.COUPONS_UPDATED`

`List<ccrz__E_Coupon__c>` of the updated coupon records.

SEE ALSO:

[Inserting and Updating Records](#)

## `ccrz.ccLogicCouponUpdateUse.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicCouponUpdateUse.performDML` method completes.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApiCoupon.COUPONLIST`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicCouponUpdateUse.COUPON_RETURN_DATA`

## ccrz.ccLogicI18NGetAvailableLocales

Define the business logic for returning a storefront's enabled locales.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiI18N.getAvailableLocales](#)

### Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method references the storefront settings and returns the list of the storefront's enabled locales.

SEE ALSO:

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

### ccrz.ccLogicI18NGetAvailableLocales.process

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method references the storefront settings and returns the list of the storefront's enabled locales.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiI18N.getAvailableLocales` method.

This method requires the following key in the input data:

### `ccrz.ccApiI18N.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Logic Performed

This method completes the following actions:


1. Queries the storefront settings object, `ccrz__E_StorefrontSettings__c`, for records that match `ccrz.ccApiI18N.PARAM_STOREFRONT`.
2. Parses the storefront settings for the list of enabled locales.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiI18N.PARAM\_AVAILABLE\_LOCALES**

Set<[String](#)> of the storefront's enabled locales.

SEE ALSO:

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

## ccrz.ccLogicI18NGetBrowserLocales

Define the business logic for determining which locales the buyer's browser supports.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiI18N.getBrowserLocales](#)

### Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method returns the list of locales that the buyer's browser supports.

SEE ALSO:

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

[Salesforce Help Supported Browsers for B2B Commerce for Visualforce Storefronts](#)

### ccrz.ccLogicI18NGetBrowserLocales.process

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method returns the list of locales that the buyer's browser supports.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiI18N.getBrowserLocales` method.

## Logic Performed

This method completes the following actions:


1. Checks the value of the `Accept-Language` request HTTP header, which specifies the locales that the user's browser supports.
2. Parses the list of returned locales, and removes any duplicates.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint.</a></p>

**ccrz.ccApiI18N.PARAM\_BROWSER\_LOCALES**

Set<[String](#)> of the browser's supported locales.

## SEE ALSO:

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

[Salesforce Help Supported Browsers for B2B Commerce for Visualforce Storefronts](#)

[Mozilla Developers Accept-Language](#)

## ccrz.ccLogicI18NGetLocale

Define the business logic for returning the current buyer's locale.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiI18N.getLocale](#)

### Methods

## IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method returns the locale for the current guest buyer or registered buyer.

## SEE ALSO:

[Salesforce Help Translate Storefront Text for Buyers in Different Locales](#)

### ccrz.ccLogicI18NGetLocale.process

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method returns the locale for the current guest buyer or registered buyer.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiI18N.getLocale` method.

This method requires the following keys in the input data:

### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.

 **Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

## Inputs (Optional)

If the input data includes the following key, this method evaluates it:

### **ccrz.ccApiI18N.PARAM\_STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Logic Performed

If the current buyer is a guest, this method completes the following actions:

1. Checks the value of the `cc1c1` URL parameter.
2. Parses the `cc1c1` URL parameter value and ensures that only one locale is specified.
3. Calls the `ccrz.ccApiI18N.getAvailableLocales` method for ensuring that the `cc1c1` URL parameter value matches one of the storefront's supported locales.
4. If `ccrz.ccApiI18N.getAvailableLocales` doesn't return a match, this method calls `ccrz.ccApiI18N.getBrowserLocales` for returning the guest buyer's browser locales, and matches `cc1c1` against the available browser locales.

If this method can't determine a locale for a guest buyer, this method sets the locale based on the site guest user record for the storefront's Experience Cloud site.


If the current buyer is registered for a storefront account, this method sets the locale based on the buyer's user record.

## Outputs

Map<[String](#), [Object](#)> that can include the following keys:

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

### **ccrz.ccApi18N.PARAM\_LOCALE**

String that specifies the current buyer's locale.

SEE ALSO:

[ccrz.ccApi18N.getAvailableLocales](#)

[ccrz.ccApi18N.getBrowserLocales](#)

[Salesforce Help Translate Storefront Text for Buyers in Different Locales](#)

[Salesforce Help Parameters in Storefront Page URLs](#)

## ccrz.ccLogic18NSetLocale

Define the business logic for setting a new locale for the current authenticated buyer.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiI18N.setLocale](#)

## Methods

### IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method references the storefront settings and returns the list of the storefront's enabled locales.

### SEE ALSO:

[Salesforce Help Translate Storefront Text for Buyers in Different Locales](#)

## ccrz.ccLogicI18NSetLocale.process

Overrides the base `ccrz.ccLogic.process` method and defines all the business logic for this logic service provider. This method references the storefront settings and returns the list of the storefront's enabled locales.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiI18N.setLocale` method.

This method requires the following keys in the input data:

### **ccrz.ccApiI18N.PARAM\_LOCALE**

String that specifies one of your [storefront's supported locales](#)—such as `ja_JP`— or generic locale—such as `ja`—to set for the user.

## Inputs (Optional)

If the input data includes the following key, this method evaluates it:

### **ccrz.ccApiI18N.PARAM\_STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Logic Performed

This method completes the following actions:


1. Passes `ccrz.ccApiI18N.PARAM_LOCALE` as input to `ccrz.ccApiI18N.getAvailableLocales` method for evaluating whether the specified locale matches one of the storefront's supported locales.
2. Updates the `LocaleSidKey` field on the buyer's corresponding [user record](#).

## Outputs

Map<[String](#), [Object](#)> that can include the following keys:

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

### **ccrz.ccApiI18N.PARAM\_LOCALE**

String that specifies the set locale for the current buyer.

 **Note:** If the method couldn't set the locale successfully, this key's value is null.

SEE ALSO:

[User \(B2B Commerce for Visualforce\)](#)

[ccrz.ccApiI18N.getAvailableLocales](#)

[Salesforce Help Translate Storefront Text for Buyers in Different Locales](#)

## ccrz.ccLogicOrderAuthCancel

Define the business logic for determining whether a buyer can cancel an order.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiOrder.canCancel](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## `ccrz.ccLogicOrderAuthCancel.process`

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiOrder.canCancel` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicOrderAuthCancel` class in the following order:

1. `processInput`
2. `prepReturn`

This method creates a `ccrz.cc_bean_MockOrder` and passes it as input for the `ccrz.cc_api_OutboundOrderCancel.eligibleForCancellation` method's `ccrz.cc_api_OutboundOrderCancel.ORDER_DATA_PARAM` input.

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed. The return data also includes the result of the `ccrz.cc_api_OutboundOrderCancel.eligibleForCancellation` method.

SEE ALSO:

[ccrz.cc\\_api\\_OutboundOrderCancel.eligibleForCancellation](#)

## ccrz.ccLogicOrderAuthCancel.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderAuthCancel.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### **ccrz.ccApiOrder.PARAM\_ORDER\_STATUS**

String that specifies the status of the order to pass to the

`ccrz.cc_api_OutboundOrderCancel.eligibleForCancellation` extension point method.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### **ccrz.ccApiOrder.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

SEE ALSO:

[ccrz.cc\\_api\\_OutboundOrderCancel.eligibleForCancellation](#)

## ccrz.ccLogicOrderAuthCancel.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicOrderAuthCancel.process` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicOrderCancel

Define the business logic for changing the status of an order to `Cancelled`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiOrder.cancel`



## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [fetchOrder](#)

Calls the `ccrz.ccApiOrder.fetch` method for returning the order to cancel.

#### [updateOrder](#)

Executes the DML `update` operation on the order record where the `ccrz.ccLogicOrderCancel.fetchCart` method set the `ccrz__OrderStatus__c` field to `Cancelled`.

#### [refetch](#)

Calls the `ccrz.ccApiOrder.fetch` method for fetching the order record after it's updated.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicOrderCancel.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiOrder.cancel` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicOrderCancel` class in the following order:

1. `processInput`
2. `fetchOrder`
3. `updateOrder`
4. `refetch`
5. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicOrderCancel.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)


This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderCancel.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID**  
Encrypted ID of the order to cancel.

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID\_LIST**

Set<String> that includes the encrypted ID of only one order to cancel.


 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID**

Salesforce ID of the order to cancel.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID\_LIST**

Set<String> that includes the Salesforce ID of only one order to cancel.

 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The [ccrz.ccLogicOrderCancel.prepReturn](#) method removes this intermediate key from the working map.

## ccrz.ccLogicOrderCancel.fetchOrder

Calls the `ccrz.ccApiOrder.fetch` method for returning the order to cancel.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchOrder (Map<String, Object>)
```

## Inputs (Required)

This method accepts the Map<String, Object> from the [ccrz.ccLogicOrderCancel.processInput](#) method.


This method executes only if the working map includes one of the following keys:

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID**

Encrypted ID of the order to cancel.

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID\_LIST**

Set<String> that includes the encrypted ID of only one order to cancel.


 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID**

Salesforce ID of the order to cancel.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID\_LIST**

Set<String> that includes the Salesforce ID of only one order to cancel.

 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

## Logic Performed

This method completes the following actions:


1. Calls the `ccrz.ccApiOrder.fetch` method to query order records.
2. Calls the `ccrz.ccApiOrder.canCancel` to determine whether the order is eligible for cancellation.
3. If the order is eligible for cancellation, sets the order's `ccrz__OrderStatus__c` field to `Cancelled`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiOrder.ORDERLIST**

List<Map<String, Object>>, where the `Map<String, Object>` represents the `ccrz__E_Order__c` record that the `ccrz.ccApiOrder.fetch` method returns.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz.ccApiOrder.fetch](#)

[ccrz.ccApiOrder.canCancel](#)

[Salesforce Help Allow a Buyer to Manage Their Orders on the My Account Page](#)

## ccrz.ccLogicOrderCancel.updateOrder

Executes the DML `update` operation on the order record where the `ccrz.ccLogicOrderCancel.fetchCart` method set the `ccrz__OrderStatus__c` field to `Cancelled`.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> updateOrder (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderCancel.fetchOrder` method.

This method requires the following key in the working map:

#### **ccrz.ccApiOrder.ORDERLIST**

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the `ccrz__E_Order__c` record where the `ccrz__OrderStatus__c` field changed to `Cancelled`.



**Note:** Even though this key specifies a list, the method evaluates only the first item in the list.

### Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[Inserting and Updating Records](#)

## ccrz.ccLogicOrderCancel.refetch

Calls the `ccrz.ccApiOrder.fetch` method for fetching the order record after it's updated.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicOrderCancel.updateOrder` method.

This method requires the following key in the working map:

### **ccrz.ccApi.SIZING**

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```



**Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiOrder.ORDERLIST**

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the updated `ccrz__E_Order__c` record.

SEE ALSO:

[ccrz.ccApiOrder.fetch](#)

## ccrz.ccLogicOrderCancel.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicOrderCancel.refetch` method.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccApiOrder.ORDERLIST`
- `ccrz.ccApiOrder.PARAM_ORDER_ID`
- `ccrz.ccApiOrder.PARAM_ORDER_ENCID`
- `ccrz.ccApiOrder.PARAM_ORDER_ID_LIST`
- `ccrz.ccApiOrder.PARAM_ORDER_ENCID_LIST`

## ccrz.ccLogicOrderHistory

Define the business logic for searching an account's order history, and fetching order records that match the search terms. This class also replaces the legacy `ccrz.cc_hk_Order.fetchOrderHistory` method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiOrder.fetchOrderHistory](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

#### [processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

#### [fetchHistory](#)

Calls the `ccrz.ccApiOrder.search`, `ccrz.ccApiOrder.fetch`, and `ccrz.ccApiOrder.fetchOrderItems` methods for searching order history for specific search terms and returning data for matching orders.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### SEE ALSO:

[ccrz.cc\\_hk\\_Order.fetchOrderHistory](#)

[Salesforce Help Allow a Buyer to Manage Their Orders on the My Account Page](#)

## ccrz.ccLogicOrderHistory.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiOrder.fetchOrderHistory` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicOrderHistory` class in the following order:

1. `processInput`
2. `fetchHistory`
3. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicOrderHistory.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderHistory.process` method.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### `ccrz.ccApiOrder.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### `ccrz.ccLogic.ORIGINAL_INPUT`

`Map<String, Object>` of the original, unmodified input data.

## `ccrz.ccLogicOrderHistory.fetchHistory`

Calls the `ccrz.ccApiOrder.search`, `ccrz.ccApiOrder.fetch`, and `ccrz.ccApiOrder.fetchOrderItems` methods for searching order history for specific search terms and returning data for matching orders.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchHistory (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderHistory.processInput` method.

This method requires either of the following keys in the working map:

**ccrz.ccApiOrder.PARAM\_ORDER\_ID\_LIST**

Set<String> of Salesforce IDs for the orders to query.

**ccrz.ccService.SEARCHSTRING**

String to use for searching order records.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccApi.API\_SIZING**

Map<String, Object> that specifies the [scope of data that this method returns](#), including which fields the method returns for each record that matches the query.

**ccrz.ccApiOrder.PARAM\_ORDERLIMIT**

Integer that specifies the maximum number of order records to query at one time.

**ccrz.ccApiOrder.PARAM\_ORDER\_ITEM\_LIMIT**

Integer that specifies the maximum number of order item records to query at one time.

## Logic Performed

This method completes the following actions:


1. If the input map specifies `ccrz.ccApiOrder.PARAM_ORDERLIMIT` or `ccrz.ccApiOrder.PARAM_ORDER_ITEM_LIMIT`, this method calls [ccrz.ccApiOrder.fetch](#) and [ccrz.ccApiOrder.fetchOrderItems](#) with the specified query limits.
2. If the input map doesn't specify `ccrz.ccApiOrder.PARAM_ORDERLIMIT` or `ccrz.ccApiOrder.PARAM_ORDER_ITEM_LIMIT`, this method attempts the following actions:
  - a. Passes `ccrz.ccService.SEARCHSTRING` as input for [ccrz.ccApiOrder.search](#).
  - b. Aggregates the search's `ccrz.ccService.SEARCHRESULTS` output with this method's `ccrz.ccApiOrder.PARAM_ORDER_ID_LIST` input, and passes the full list of IDs as input for [ccrz.ccApiOrder.fetch](#).
  - c. If the search returned no results, passes `ccrz.ccApiOrder.PARAM_ORDER_ID_LIST` as input for [ccrz.ccApiOrder.fetch](#).

## Outputs

This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

This method's output also includes the output of `ccrz.ccApiOrder.search`, `ccrz.ccApiOrder.fetch`, and `ccrz.ccApiOrder.fetchOrderItems`, if applicable.

#### SEE ALSO:

[ccrz.ccApiOrder.fetch](#)

[ccrz.ccApiOrder.fetchOrderItems](#)

[ccrz.ccApiOrder.search](#)

[Salesforce Help Allow a Buyer to Manage Their Orders on the My Account Page](#)

## ccrz.ccLogicOrderHistory.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicOrderHistory.fetchHistory` method.

### Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicOrderGetOrdersToSplit

Define the business logic for determining which orders to include in the `ccrz.ccSplitOrderJob` batch processing class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiOrder.getOrdersToSplit](#)

## Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, and defines the entire logic for the class.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz.ccSplitOrderJob](#)

## ccrz.ccLogicOrderGetOrdersToSplit.process

Overrides the base `ccrz.ccLogic.process` method, and defines the entire logic for the class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiOrder.getOrdersToSplit` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### **ccrz.ccApiOrder.PARAM\_STOREFRONT**

String that specifies the name of the storefront to split orders for.

## Logic Performed

This method calls `Database.getQueryLocator` with a query that selects `ccrz__E_Order__c` records where:

- The `ccrz__Storefront__c` field matches `ccrz.ccApiOrder.PARAM_STOREFRONT`.
- The `ccrz__SplitProcessed__c` field is `FALSE`.

## Outputs

`Map<String, Object>` that includes the following key:

### **ccrz.ccApiOrder.SPLITSELECTOR**

`QueryLocator` for the `ccrz.ccSplitOrderJob` batch processing class to execute against.

## ccrz.ccLogicOrderReOrder

Define the business logic for copying a completed order and its order items to a new cart and cart items. This class also replaces the legacy `ccrz.cc_hk_Order.reorder` method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiOrder.reorder`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [fetchOrder](#)

Calls the `ccrz.ccApiOrder.fetch` method for returning the order to copy.

#### [fetchAddresses](#)

Calls the `ccrz.ccApiAddress.fetch` method for returning billing and shipping addresses from the order and its corresponding order item groups.

#### [processAddresses](#)

Assigns a new external ID to each billing and shipping address copied from the original order. This method also returns a map that associates each original address' Salesforce ID to its new external ID.

#### [insertAddresses](#)

Calls the `ccrz.ccApiAddress.createAddress` method, which executes the DML `insert` operation on the new billing and shipping addresses. This method also updates the map of old-to-new addresses with the Salesforce IDs of the inserted address records.

#### [processOrderData](#)

Copies data from the original order, order items, and order item groups to the corresponding cart, cart items, and cart item groups. This method creates the cart-related sObjects and assigns new external IDs.

#### [createCart](#)

Calls the `ccrz.ccApiCart.create` method for creating the `ccrz__E_Cart__c` sObject.

#### [insertCartItemGroups](#)

Executes the DML `insert` operation on the new `ccrz__E_CartItemGroup__c` records.

#### [insertCartItems](#)

Executes the DML `insert` operation on the new `ccrz__E_CartItem__c` records and DML `upsert` operation on child cart items.

#### [refetch](#)

Calls the `ccrz.ccApiCart.fetch` method for fetching the cart record after it's created.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

[ccrz.ccLogicOrderReOrder7HK Example Extension Class](#)

The managed package provides an example extension class, `ccrz.ccLogicOrderReOrder7HK`, that you can use as a starting point for extending `ccrz.ccLogicOrderReOrder` to reference the legacy `ccrz.cc_hk_Order` class.

## SEE ALSO:

[ccrz.cc\\_hk\\_Order.reorder](#)

[Salesforce Help Allow a Buyer to Manage Their Orders on the My Account Page](#)

[Salesforce Help Show the Reorder Widget](#)

## ccrz.ccLogicOrderReOrder.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiOrder.reorder` method.

### Logic Performed

This method invokes other methods of the `ccrz.ccLogicOrderReOrder` class in the following order:


1. `processInput`
2. `fetchOrder`
3. `fetchAddresses`
4. `processAddresses`
5. `insertAddresses`
6. `processOrderData`



7. `createCart`
8. `insertCartItemGroups`
9. `insertCartItems`
10. `refetch`
11. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicOrderReOrder.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderReOrder.process` method.


This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

### **ccrz.ccApiOrder.PARAM\_ORDER\_ENCID**

Encrypted ID of the order to copy.

### **ccrz.ccApiOrder.PARAM\_ORDER\_ENCID\_LIST**

`Set<String>` that includes the encrypted ID of only one order to copy.


 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID**

Salesforce ID of the order to copy.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID\_LIST**

Set<String> that includes the Salesforce ID of only one order to copy.

 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The [ccrz.ccLogicOrderReOrder.prepReturn](#) method removes this intermediate key from the working map.

## ccrz.ccLogicOrderReOrder.fetchOrder

Calls the `ccrz.ccApiOrder.fetch` method for returning the order to copy.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchOrder (Map<String, Object>)
```

## Inputs (Required)

This method accepts the Map<String, Object> from the [ccrz.ccLogicOrderReOrder.processInput](#) method.


This method executes only if the working map includes one of the following keys:

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID**

Encrypted ID of the order to copy.

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID\_LIST**

Set<String> that includes the encrypted ID of only one order to copy.


 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID**

Salesforce ID of the order to copy.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID\_LIST**

Set<String> that includes the Salesforce ID of only one order to copy.

 **Note:** Even though this key specifies a set, the method evaluates only the first ID in the set.

## Logic Performed

This method calls the `ccrz.ccApiOrder.fetch` method to query the order, order items, and order item groups records to copy. By default, this query applies the following [data size request parameters](#):


```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XL,
        ccrz.ccApi.SZ_ASSC => false,
        ccrz.ccApi.SZ_SKIPTRZ => true,
        ccrz.ccApi.SZ_REL => new List<String>{'E_OrderItems__r', 'E_OrderItemGroups__r'}
    }
}
```

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

**ccrz.ccApiOrder.ORDERITEMSBYID**

Map<String, List<Map<String, Object>>>, keyed by parent `ccrz__E_Order__c`, where each Map<String, Object> represents a `ccrz__E_OrderItem__c` record associated with the order.

**ccrz.ccApiOrder.ORDERLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the `ccrz__E_Order__c` record that the `ccrz.ccApiOrder.fetch` method returns.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[ccrz\\_\\_E\\_OrderItemGroup\\_\\_c](#)

[ccrz.ccApiOrder.fetch](#)

## ccrz.ccLogicOrderReOrder.fetchAddresses

Calls the `ccrz.ccApiAddress.fetch` method for returning billing and shipping addresses from the order and its corresponding order item groups.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchAddresses (Map<String, Object>)
```

### Inputs (Required)

This method accepts the Map<String, Object> from the `ccrz.ccLogicOrderReOrder.fetchOrder` method.

This method requires the following keys in the working map:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The previous method in the logic chain completed successfully. This method executes.
<i>false</i>	This previous method in the logic chain failed. Execution skips this method, and continues to the next method in the logic chain.

**ccrz.ccApiOrder.ORDERLIST**

List<Map<String, Object>>, where the Map<String, Object> represents the [ccrz\\_\\_E\\_Order\\_\\_c](#) record whose billing and shipping addresses this method queries.

**Logic Performed**

This method completes the following actions:

1. Evaluates the order's [ccrz\\_\\_BillTo\\_\\_c](#) and [ccrz\\_\\_ShipTo\\_\\_c](#) fields, and adds the IDs of the referenced [ccrz\\_\\_E\\_ContactAddr\\_\\_c](#) records to a Set<String> of address IDs to query.
2. Loops through the [ccrz\\_\\_E\\_OrderItemGroup\\_\\_c](#) records associated with the order, and adds their [ccrz\\_\\_ShipTo\\_\\_c](#) addresses to the Set<String> of address IDs to query.
3. Passes the Set<String> of address IDs to [ccrz.ccApiAddress.fetch](#).

**Outputs**

This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccApiAddress.ADDRESSLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a [ccrz\\_\\_E\\_ContactAddr\\_\\_c](#) record that describes a billing or shipping address associated with the order or its order item groups.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz\\_\\_E\\_OrderItemGroup\\_\\_c](#)

[ccrz.ccApiAddress.fetch](#)

**ccrz.ccLogicOrderReOrder.processAddresses**

Assigns a new external ID to each billing and shipping address copied from the original order. This method also returns a map that associates each original address' Salesforce ID to its new external ID.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processAddresses (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderReOrder.fetchAddresses` method.

This method requires the following keys in the working map:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The previous method in the logic chain completed successfully. This method executes.
<code>false</code>	This previous method in the logic chain failed. Execution skips this method, and continues to the next method in the logic chain.

### `ccrz.ccApiAddress.ADDRESSLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c` record that describes a billing or shipping address associated with the order or its order item groups.

## Logic Performed

For each address in `ccrz.ccApiAddress.ADDRESSLIST`, this method completes the following actions:

1. Removes the existing `sfid` value.
2. Assigns a new, randomized `contactAddrId` value using the `Crypto.getRandomInteger` Apex method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiAddress.ADDRESSLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c` record, updated with a new `contactAddrId` value.

### 'origAddressToNewId'

`Map<String, Object>` that associates each address' original `sfid` to its new `contactAddrId`.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

## `ccrz.ccLogicOrderReOrder.insertAddresses`

Calls the `ccrz.ccApiAddress.createAddress` method, which executes the DML `insert` operation on the new billing and shipping addresses. This method also updates the map of old-to-new addresses with the Salesforce IDs of the inserted address records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> insertAddresses (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderReOrder.processAddresses` method.

This method requires the following keys in the working map:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The previous method in the logic chain completed successfully. This method executes.
<code>false</code>	This previous method in the logic chain failed. Execution skips this method, and continues to the next method in the logic chain.

### `ccrz.ccApiAddress.ADDRESSLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_ContactAddr__c` record to insert.

### `'origAddressToNewId'`

`Map<String, Object>` that associates each address' original `sfid` to its new `contactAddrId`.

## Logic Performed

This method completes the following actions:

1. Passes `ccrz.ccApiAddress.ADDRESSLIST` as input for the `ccrz.ccApiAddress.createAddress` method. By default, this method applies the following [data size request parameters](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true,
```

```

        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}

```

2. Calls `ccrz.ccApiAddress.fetch`—because the `ccrz.ccApiAddress.createAddress` input map specified `ccrz.ccApi.SZ_REFETCH => true`—to return the inserted addresses.
3. Evaluates the inserted addresses returned from `ccrz.ccApiAddress.fetch`, and overwrites the existing `'origAddressToNewId'` map with a new map of each address' original Salesforce ID to its new Salesforce ID.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiAddress.ADDRESSLIST`

`List<Map<String, Object>>`, where each `Map<String, Object>` represents an inserted `ccrz__E_ContactAddr__c` record.

### `'origAddressToNewId'`

`Map<String, Object>` that associates each address' original `sfid` to its new `sfid`.

SEE ALSO:

[ccrz.ccApiAddress.createAddress](#)

[ccrz.ccApiAddress.fetch](#)

[Inserting and Updating Records](#)

## `ccrz.ccLogicOrderReOrder.processOrderData`

Copies data from the original order, order items, and order item groups to the corresponding cart, cart items, and cart item groups. This method creates the cart-related sObjects and assigns new external IDs.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processOrderData (Map<String, Object>)
```



## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderReOrder.insertAddresses` method.

This method requires the following keys in the working map:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The previous method in the logic chain completed successfully. This method executes.
<code>false</code>	This previous method in the logic chain failed. Execution skips this method, and continues to the next method in the logic chain.

### `ccrz.ccApiOrder.ORDERITEMSBYID`

`Map<String, List<Map<String, Object>>>`, keyed by parent `ccrz__E_Order__c`, where each `Map<String, Object>` represents a `ccrz__E_OrderItem__c` record to copy.

### `ccrz.ccApiOrder.ORDERLIST`

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the `ccrz__E_Order__c` record to copy.

### `'origAddressToNewId'`

`Map<String, Object>` that associates each billing and shipping address' original `sfid` to its new `sfid`.

## Logic Performed

This method completes the following actions:

1. Evaluates `ccrz.ccApiOrder.ORDERITEMSBYID` and creates an intermediate key, `'orderItemsById'`, for organizing the original order's items into major and minor line items.
2. Creates a `Map<String, Object>` for the new `ccrz__E_Cart__c` record, and assigns a randomized `cartId` value using the `Crypto.getRandomInteger` Apex method.
3. Evaluates `'origAddressToNewId'` and assigns the cart's `ccrz__BillTo__c` and `ccrz__ShipTo__c` contact addresses.
4. Evaluates the order's `ccrz__E_OrderItemGroup__c` records, and creates corresponding `ccrz__E_CartItemGroup__c` records. This method assigns each group a randomized `cartItemGroupId` value using the `Crypto.getRandomInteger` Apex method.
5. Evaluates the order's `ccrz__E_OrderItem__c` records, and creates corresponding `ccrz__E_CartItem__c` records. This method assigns each item a randomized `cartItemId` value using the `Crypto.getRandomInteger` Apex method.



**Note:** This method sets each new cart item's `ccrz__Price__c` and `ccrz__SubAmount__c` values to `0.00` initially. The `ccrz.ccLogicCartPrice` class handles repricing the cart items.

6. Prepares lists of `ccrz__E_Cart__c`, `ccrz__E_CartItem__c`, and `ccrz__E_CartItemGroup__c` sObjects to insert.
7. Assigns cart items that are minor line items, such as component products of composite products (bundles, kits, and so on), to a separate list for upsert.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiCart.CART_OBJLIST`

List<`ccrz__E_Cart__c`> that contains the cart record to create.

### '`cartItemsToInsert`'

List<`ccrz__E_CartItem__c`> of sObjects, with external ID set, ready to insert.

### '`groupsToInsert`'

List<`ccrz__E_CartItemGroup__c`> of sObjects, with external ID set, ready to insert.

### '`upsertList`'

List<`ccrz__E_CartItem__c`> of sObjects for child cart items, such as component products for composite products.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[ccrz\\_\\_E\\_OrderItemGroup\\_\\_c](#)

## `ccrz.ccLogicOrderReOrder.createCart`

Calls the `ccrz.ccApiCart.create` method for creating the `ccrz__E_Cart__c` sObject.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createCart (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderReOrder.processOrderData` method.

This method requires the following keys in the working map:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The previous method in the logic chain completed successfully. This method executes.
<code>false</code>	This previous method in the logic chain failed. Execution skips this method, and continues to the next method in the logic chain.

**ccrz.ccApiCart.CART\_OBJLIST**

List<[ccrz\\_\\_E\\_Cart\\_\\_c](#)> that contains the cart record to create.

## Logic Performed

This method passes `ccrz.ccApiCart.CART_OBJLIST` as in the input for [ccrz.ccApiCart.create](#).

## Outputs


This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccApi.MESSAGES**

List<[ccrz.cc\\_bean\\_Message](#)>, where each `ccrz.cc_bean_Message` contains error information sent from the server to the client. Typically, these messages are included only when `ccrz.ccApi.SUCCESS` is FALSE.

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the created cart.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the created cart.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz.ccApiCart.create](#)

## ccrz.ccLogicOrderReOrder.insertCartItemGroups

Executes the DML `insert` operation on the new `ccrz__E_CartItemGroup__c` records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> insertCartItemGroups (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderReOrder.createCart` method.

This method requires the following keys in the working map:

#### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The previous method in the logic chain completed successfully. This method executes.
<i>false</i>	This previous method in the logic chain failed. Execution skips this method, and continues to the next method in the logic chain.

#### **ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the created cart.

#### **'groupsToInsert'**

List<`ccrz__E_CartItemGroup__c`> of sObjects, with external ID set, ready to insert.

### Logic Performed

This method assigns the `ccrz.ccApiCart.CART_ID` for the `ccrz__Cart__c` field on each inserted `ccrz__E_CartItemGroup__c`.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_CartItemGroup\\_\\_c](#)  
[Inserting and Updating Records](#)

## ccrz.ccLogicOrderReOrder.insertCartItems

Executes the DML `insert` operation on the new `ccrz__E_CartItem__c` records and DML `upsert` operation on child cart items.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> insertCartItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderReOrder.insertCartItemGroups` method.

This method requires the following keys in the working map:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<code>true</code>	The previous method in the logic chain completed successfully. This method executes.
<code>false</code>	This previous method in the logic chain failed. Execution skips this method, and continues to the next method in the logic chain.

**ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the created cart.

**'cartItemsToInsert'**

List<ccrz\_\_E\_CartItem\_\_c> of sObjects, with external ID set, ready to insert.

**'upsertList'**

List<ccrz\_\_E\_CartItem\_\_c> of sObjects for child cart items, such as component products for composite products.

## Logic Performed

This method assigns the `ccrz.ccApiCart.CART_ID` for the `ccrz__Cart__c` field on each inserted `ccrz__E_CartItem__c`.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[Inserting and Updating Records](#)

[Upserting Records](#)

## ccrz.ccLogicOrderReOrder.refetch

Calls the `ccrz.ccApiCart.fetch` method for fetching the cart record after it's created.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch (Map<String, Object>)
```

## Inputs (Required)


This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicOrderReOrder.insertCartItems` method.

This method requires the following keys in the working map:

### **ccrz.ccApi.SIZING**

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true` for the `ccrz.ccApiCart.ENTITYNAME`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiCart.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The previous method in the logic chain completed successfully. This method executes.
<i>false</i>	This previous method in the logic chain failed. Execution skips this method, and continues to the next method in the logic chain.

### **ccrz.ccApiCart.CART\_ID**

String that specifies the Salesforce ID of the created cart.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiCart.CART\_OBJLIST**

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the created `ccrz__E_Cart__c` record that the `ccrz.ccApiCart.fetch` method returned.

SEE ALSO:

[ccrz.ccApiCart.fetch](#)

## ccrz.ccLogicOrderReOrder.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicOrderReOrder.refetch` method.

This method requires the following keys in the working map:

### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
<i>true</i>	The previous method in the logic chain completed successfully. This method executes.
<i>false</i>	This previous method in the logic chain failed. Execution skips this method, and continues to the next method in the logic chain.

### **ccrz.ccApiCart.CART\_ENCID**

String that specifies the encrypted ID of the created cart.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`
- `'cartItemsToInsert'`
- `'groupsToInsert'`
- `'orderItemsById'`
- `'upsertList'`

## ccrz.ccLogicOrderReOrder7HK Example Extension Class

The managed package provides an example extension class, `ccrz.ccLogicOrderReOrder7HK`, that you can use as a starting point for extending `ccrz.ccLogicOrderReOrder` to reference the legacy `ccrz.cc_hk_Order` class.



## Compatibility

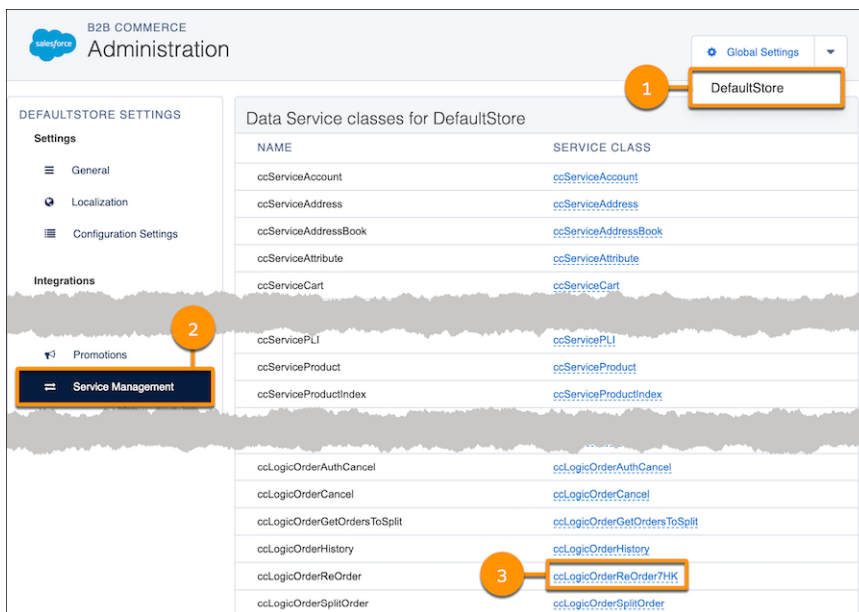
This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

This extension implements the following custom logic:

1. Overrides the default `ccrz.ccLogicOrderReOrder.process` method to call only the following inner methods:
  - `processInput`
  - `createCart`
  - `prepReturn`
2. Overrides the default `processInput` method to set the required order ID inputs.
3. Overrides the default `createCart` method to call `ccrz.cc_hk_Order.reorder`.
4. Overrides the default `prepReturn` method to remove intermediate keys.

To enable this extension, update your storefront settings in CC Admin. Select your storefront (1), and then select **Service Management** in the Integrations section of the Storefront Settings menu (2). Update the `ccLogicOrderReOrder` class (3) with `ccLogicOrderReOrder7HK`.





**Tip:** For more help with extending a default logic service provider, including considerations and conventions, see [Extend or Override a Default Logic Service Provider](#).

SEE ALSO:

[ccrz.cc\\_hk\\_Order](#)

## ccrz.ccLogicOrderRestore

Define the business logic for reopening a cart after the corresponding order fails when attempting to place the order asynchronously. This cart's status resets to Open and the cart becomes active. This logic service provider can only restore an order with a status of Error and that hasn't been restored previously.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

### Global API Caller

[ccrz.ccApiOrder.restore](#)

### Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

[processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

[fetchOrder](#)

Calls the `ccrz.ccApiOrder.fetch` method for returning the order that failed to place.

[reOpenCart](#)

Calls the `ccrz.ccApiCart.fetch` method for returning the cart that corresponds to the failed order, and updates values on both the cart and order records.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### ccrz.ccLogicOrderRestore.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiOrder.restore` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicOrderRestore` class in the following order:

1. `processInput`
2. `fetchOrder`
3. `reOpenCart`
4. `prepReturn`

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicOrderRestore.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderCancel.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID**

String that specifies the encrypted ID of the order to restore.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID**

String that specifies the Salesforce ID of the order to restore.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

**ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicOrderRestore.prepReturn` method removes this intermediate key from the working map.

This method also instantiates the following return data keys:

**ccrz.ccApi.SUCCESS**

Boolean that's assigned an intermediate value of `true`. Subsequent methods in the logic chain update this value.

## ccrz.ccLogicOrderRestore.fetchOrder

Calls the `ccrz.ccApiOrder.fetch` method for returning the order that failed to place.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> fetchOrder (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicOrderRestore.processInput` method completes.

This method requires either of the following keys in the working map:

**ccrz.ccApiOrder.PARAM\_ORDER\_ENCID**

String that specifies the encrypted ID of the order to restore.

**ccrz.ccApiOrder.PARAM\_ORDER\_ID**

String that specifies the Salesforce ID of the order to restore.

## Logic Performed

This method completes the following actions:

1. Passes `ccrz.ccApiOrder.PARAM_ORDER_ENCID` or `ccrz.ccApiOrder.PARAM_ORDER_ID` as input for the `ccrz.ccApiOrder.fetch` method.
2. Stores the fetched `ccrz__E_Order__c` in the `ccrz.ccApiOrder.ORDERLIST` intermediate key.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiOrder.ORDERLIST`

`List<ccrz__E_Order__c>` that contains the order that failed to place.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)  
[ccrz.ccApiOrder.fetch](#)

## ccrz.ccLogicOrderRestore.reOpenCart

Calls the `ccrz.ccApiCart.fetch` method for returning the cart that corresponds to the failed order, and updates values on both the cart and order records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> reOpenCart (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` after the `ccrz.ccLogicOrderRestore.fetchOrder` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiOrder.ORDERLIST`

`List<ccrz__E_Order__c>` that contains the order that failed to place.



**Note:** By default, this method expects only one order record in the list.

## Logic Performed

This method completes the following actions:

1. Parses `ccrz.ccApiOrder.ORDERLIST` for the first `ccrz__E_Order__c` record included in the list.
2. Evaluates whether the order satisfies the following criteria:
  - The order isn't null.
  - The order's `ccrz__CartRestored__c` field is `false`, which indicates that the order hasn't already been restored previously.
  - The order's `ccrz__OrderStatus__c` field value is `Error`.
  - The order's `ccrz__OriginatedCart__c` field specifies some value, which indicates that the order was placed from a cart.
3. Passes the order's `ccrz__OriginatedCart__c` as the `ccrz.ccApiCart.CART_ID` input for the `ccrz.ccApiCart.fetch` method.
4. Parses the fetched cart data for the first `ccrz__E_Cart__c` record.
5. Calls the `ccrz.ccApiCart.revise` method to reset the cart's `ccrz__CartStatus__c` field to `Open`.
6. Updates the order's `ccrz__CartRestored__c` field to `true`.
7. Stores the encrypted ID of the reopened cart in the 'originatedCart' output.
8. Calls the `ccrz.ccApiCart.setActive` method to make the reopened cart active.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### 'originatedCart'

String that specifies the encrypted ID of the reopened cart that corresponds to the order that failed to place.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)  
[ccrz\\_\\_E\\_Order\\_\\_c](#)  
[ccrz.ccApiCart.fetch](#)  
[ccrz.ccApiCart.revise](#)  
[ccrz.ccApiCart.setActive](#)

## ccrz.ccLogicOrderRestore.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderRestore.reOpenCart` method completes.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicOrderSplitOrder

Define the business logic for splitting orders by seller. To split orders by other criteria, extend this logic service provider class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiOrder.splitOrder](#)

## Methods

IN THIS SECTION:

### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [retrieveOrderData](#)

Calls the `ccrz.ccApiOrder.fetch` method for returning the orders to split.

#### [prepareSplits](#)

Splits order items into a map that describes the split criteria, and builds a map of seller IDs and seller representative users.

#### [retrieveAdditionalData](#)

Calls the `ccrz.ccApiAddress.fetch` method to return the `ccrz__E_ContactAddr__c` records to set for each split order's billing and shipping addresses. To fetch other data to apply to split orders, override this method in your subscriber extension code.

#### [createNewOrders](#)

Creates `ccrz__E_Order__c` records based on the specified split order criteria.

#### [copyOrder](#)

Copies the original order data to a new `ccrz__E_Order__c` record.

#### [setOrderOwner](#)

Assigns an owner to a new order that the `ccrz.ccLogicOrderSplitOrder.copyOrder` method created.

#### [updateOrders](#)

Executes the DML `update` operation on the original order records that the `ccrz.ccLogicOrderSplitOrder.createNewOrders` method split into child orders.

#### [createAdditionalData](#)

Calls the `ccrz.ccApiAddress.createAddress` method to create `ccrz__E_ContactAddr__c` records from the copies of the original order's billing and shipping addresses. To create other data to apply to split orders, override this method in your subscriber extension code.

#### [insertNewOrders](#)

Assigns billing and shipping addresses on the new `ccrz__E_Order__c` records, and executes the DML `insert` operation on the `ccrz__E_Order__c` records.

#### [createNewOrderItems](#)

Creates `ccrz__E_OrderItem__c` records for the new orders, based on the specified split order criteria.

#### [copyOrderItem](#)

Copies an original order item data to a new `ccrz__E_OrderItem__c` record.

#### [insertNewOrderItems](#)

Executes the DML `insert` operation on the new `ccrz__E_OrderItem__c` records.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

SEE ALSO:

[ccrz.ccSplitOrderJob](#)

## `ccrz.ccLogicOrderSplitOrder.process`

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiOrder.splitOrder` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicOrderSplitOrder` class in the following order:

1. `processInput`
2. `retrieveOrderData`
3. `prepareSplits`
4. `retrieveAdditionalData`
5. `createNewOrders`, which calls into `copyOrder`, which calls into `setOrderOwner`
6. `updateOrders`
7. `createAdditionalData`
8. `insertNewOrders`
9. `createNewOrderItems`, which calls into `copyOrderItem`
10. `insertNewOrderItems`
11. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.



**Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicOrderSplitOrder.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicOrderSplitOrder.process` method.

This method checks that the input map specifies the following keys, and returns a `ccrz.ccApi.MissingInputException` if any key is missing:

**ccrz.ccApiOrder.PARAM\_ORDER\_ID\_LIST**

`Set<String>` that includes the Salesforce IDs of the orders to split.

**ccrz.ccApiOrder.PARAM\_STOREFRONT**

String that specifies the name of the storefront to split orders for.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicOrderSplitOrder.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicOrderSplitOrder.retrieveOrderData

Calls the `ccrz.ccApiOrder.fetch` method for returning the orders to split.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> retrieveOrderData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderSplitOrder.processInput` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiOrder.PARAM\_ORDER\_ID\_LIST**  
`Set<String>` that includes the Salesforce IDs of the orders to split.

**ccrz.ccApiOrder.PARAM\_STOREFRONT**  
String that specifies the name of the storefront to split orders for.

## Logic Performed

This method calls the `ccrz.ccApiOrder.fetch` method to query the order records to split, including the associated order items. By default, this query applies the following [data size request parameters](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiOrder.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XL,
        ccrz.ccApi.SZ_SKIPTRZ => true,
        ccrz.ccApi.SZ_REL => new List<String>{'E_OrderItems__r'}
    }
}
```

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiOrder.ORDERITEMSBYID**  
`Map<String, List<ccrz__E_OrderItem__c>>` of the original order's items, keyed by parent `ccrz__E_Order__c`.

**ccrz.ccApiOrder.ORDERLIST**  
`List<ccrz__E_Order__c>` that the `ccrz.ccApiOrder.fetch` method returns.

SEE ALSO:

[ccrz.ccApiOrder.fetch](#)

## ccrz.ccLogicOrderSplitOrder.prepareSplits

Splits order items into a map that describes the split criteria, and builds a map of seller IDs and seller representative users.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepareSplits (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderSplitOrder.retrieveOrderData` method completes.

This method requires the following keys in the working map:

#### **ccrz.ccApiOrder.ORDERITEMSBYID**

`Map<String, List<ccrz__E_OrderItem__c>>` of the original order's items, keyed by parent `ccrz__E_Order__c`.

#### **ccrz.ccApiOrder.ORDERLIST**

`List<ccrz__E_Order__c>` of the original orders to split.

### Logic Performed

This method completes the following actions:

1. Evaluates each order and builds `ccrz.ccApiOrder.SPLORDITM`, which is a map of maps, where:

- The outer map's key is the original order ID.
- Each inner map's key is the criteria for splitting `ccrz.ccApiOrder.ORDERITEMSBYID`, such as the ID of the referenced `ccrz__E_Seller__c` from each `ccrz__E_OrderItem__c` record's `ccrz__Seller__c` field.
- Each inner map's value is the list of corresponding `ccrz__E_OrderItem__c` data.

Consider the following example, where:

- Original Order A contains two order items: one order item from Seller X and another order item from Seller Y.
- Original Order B contains three order items: two order items from Seller X and another order item from Seller Y.

The resulting `ccrz.ccApiOrder.SPLORDITM` map resembles the following serialized JSON:

```
{
  "Order A" : {
    "Seller X" : [Order Item 1],
    "Seller Y" : [Order Item 2]
  },
  "Order B" : {
    "Seller X" : [Order Item 3,Order Item 4],
    "Seller Y" : [Order Item 5]
  }
}
```

2. Passes the seller IDs associated the order items as input for the `ccrz.ccApiSeller.fetch` method.
3. Evaluates the `ccrz.ccApiSeller.SELLERLIST` output from `ccrz.ccApiSeller.fetch` and adds each seller's ID and `ccrz__DefaultRep__c` user to `ccrz.ccApiOrder.OWNERPERSPLT`, a map for other downstream methods to evaluate.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiOrder.OWNERPERSPLT`

`Map<String, String>` that associates each seller ID to its `ccrz__DefaultRep__c` user.

### `ccrz.ccApiOrder.SPLORDITM`

`Map<String, Map<String, List<ccrz__E_OrderItem__c>>>` of the split order items, keyed first by order ID and then by split criteria, such as seller ID.

SEE ALSO:

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[ccrz\\_\\_E\\_Seller\\_\\_c](#)

[ccrz.ccApiSeller.fetch](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## `ccrz.ccLogicOrderSplitOrder.retrieveAdditionalData`

Calls the `ccrz.ccApiAddress.fetch` method to return the `ccrz__E_ContactAddr__c` records to set for each split order's billing and shipping addresses. To fetch other data to apply to split orders, override this method in your subscriber extension code.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> retrieveAdditionalData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderSplitOrder.prepareSplits` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiOrder.ORDERLIST**

List<`ccrz__E_Order__c`> of the original orders to split.

### **ccrz.ccApiOrder.SPLORDITM**

Map<String, Map<String, List<`ccrz__E_OrderItem__c`>>> of the split order items, keyed first by order ID and then by split criteria, such as seller ID.

## Logic Performed

This method completes the following actions:

1. Evaluates each order in `ccrz.ccApiOrder.ORDERLIST`, and builds a list of billing and shipping address IDs to fetch.
2. Passes the list of address IDs to the `ccrz.ccApiAddress.fetch` method for returning the `ccrz__E_ContactAddr__c` records for billing and shipping addresses to set on the split orders.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiOrder.ADDRIDMAP**

Map<String, Object>, where:

- Each key is the ID of a `ccrz__E_ContactAddr__c` record.
- Each value is a Map<String, Object> represents the corresponding `ccrz__E_ContactAddr__c` record that `ccrz.ccApiAddress.fetch` returned.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz.ccApiAddress.fetch](#)

## ccrz.ccLogicOrderSplitOrder.createNewOrders

Creates `ccrz__E_Order__c` records based on the specified split order criteria.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createNewOrders (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderSplitOrder.retrieveAdditionalData` method completes.

This method requires the following keys in the working map:

### `ccrz.ccApiOrder.ADDRIDMAP`

`Map<String, Object>`, where:

- Each key is the ID of a `ccrz__E_ContactAddr__c` record for an order's billing or shipping address.
- Each value is a `Map<String, Object>` represents the corresponding `ccrz__E_ContactAddr__c` record.

### `ccrz.ccApiOrder.ORDERLIST`

`List<ccrz__E_Order__c>` of the original orders to split.

### `ccrz.ccApiOrder.OWNERPERSPLT`

`Map<String, String>` that associates each seller ID to its `ccrz__DefaultRep__c` user.

### `ccrz.ccApiOrder.SPLORDITM`

`Map<String, Map<String, List<ccrz__E_OrderItem__c>>>` of the split order items, keyed first by order ID and then by split criteria, such as seller ID.

## Logic Performed

This method evaluates the `ccrz.ccApiOrder.SPLORDITM` map and calls the `ccrz.ccLogicOrderSplitOrder.copyOrder` inner method, which creates a `ccrz__E_Order__c` record for each combination of original order ID and seller ID.

Recall the following example from the `ccrz.ccLogicOrderSplitOrder.prepareSplits` method reference, where:

- Original Order A contains two order items: one order item from Seller X and another order item from Seller Y.
- Original Order B contains three order items: two order items from Seller X and another order item from Seller Y.

The `ccrz.ccApiOrder.SPLORDITM` map resembles the following serialized JSON:

```
{
  "Order A" : {
    "Seller X" : [Order Item 1],
    "Seller Y" : [Order Item 2]
  },
  "Order B" : {
    "Seller X" : [Order Item 3,Order Item 4],
    "Seller Y" : [Order Item 5]
  }
}
```

From this map, this method creates the following child orders:

Child Order Number	Parent Order	Seller	Order Items
1	Order A	X	Order Item 1
2	Order A	Y	Order Item 2
3	Order B	X	Order Item 3, Order Item 4
4	Order B	Y	Order Item 5

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiOrder.NEWORDLST`

`List<ccrz__E_Order__c>` of the new orders that the `ccrz.ccLogicOrderSplitOrder.copyOrder` method creates.

### `ccrz.ccApiOrder.OLDADDRDATA`

`List<Map<String, Object>>`, where each `Map<String, Object>` describes a `ccrz__E_ContactAddr__c` record that the original order referenced for a billing or shipping address.



**Note:** The `ccrz.ccLogicOrderSplitOrder.copyOrder` method copies these addresses, and the `ccrz.ccLogicOrderSplitOrder.insertNewOrders` method assigns the copies to the new order records.

### `ccrz.ccApiOrder.ORDCLNMAP`

`Map<String, Map<String, ccrz__E_Order__c>>`, where:

- The outer map's key is the original order ID.
- Each inner map's key is the criteria for splitting order items: seller ID.
- Each inner map's value is the list of created child `ccrz__E_Order__c` records.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz\\_\\_E\\_Seller\\_\\_c](#)

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)



## ccrz.ccLogicOrderSplitOrder.copyOrder

Copies the original order data to a new `ccrz__E_Order__c` record.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> copyOrder (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicOrderSplitOrder.createNewOrders` method.

This method requires the following keys in the working map:

#### **ccrz.ccApiOrder.ADDRIDMAP**

`Map<String, Object>`, where:

- Each key is the ID of a `ccrz__E_ContactAddr__c` record for an order's billing or shipping address.
- Each value is a `Map<String, Object>` represents the corresponding `ccrz__E_ContactAddr__c` record.

#### **ccrz.ccApiOrder.OLDADDRDATA**

`List<Map<String, Object>>`, where each `Map<String, Object>` describes a `ccrz__E_ContactAddr__c` record that the original order referenced for a billing or shipping address.



#### Note:

- This method copies these addresses, and the `ccrz.ccLogicOrderSplitOrder.insertNewOrders` method assigns the copies to the new order records.
- If you override this method in an extension and add other address data, be sure to set each address' `ownerId` to the same value as `ccrz.ccApiOrder.PARAM_OWNERID`.

#### **ccrz.ccApiOrder.PARAM\_OWNERID**

String that specifies the owner for the new order.

#### **ccrz.ccApiOrder.SPLITID**

String that specifies the criteria to split the original order by, such as seller ID.


**ccrz.ccApiOrder.SRCORD**

`ccrz__E_Order__c`, which specifies the original order record to copy.

## Logic Performed

The `ccrz.ccLogicOrderSplitOrder.createNewOrders` method calls into this method for creating each new order. This method completes the following actions for each new order:

1. Calls the sObject `clone` method to create a duplicate of the original `ccrz__E_Order__c`.
2. Sets the following field values on the new order:

Field	Value
<code>ccrz__AdjustmentReason__c</code>	null
<code>ccrz__EncryptedId__c</code>	null
<code>ccrz__OrderNumber__c</code>	null
<code>ccrz__OrderSeller__c</code>	The seller ID.
<code>ccrz__SplitParentOrder__c</code>	The ID of <code>ccrz.ccApiOrder.SRCORD</code>
<code>ccrz__SplitProcessed__c</code>	TRUE  <b>Note:</b> When this field is FALSE, the order remains eligible for future <code>ccrz.ccSplitOrderJob</code> batch processing.

3. Calls the `ccrz.ccLogicOrderSplitOrder.setOrderOwner` method for setting the new order's owner to `ccrz.ccApiOrder.PARAM_OWNERID`.
4. Creates copies of the address data, which the `ccrz.ccLogicOrderSplitOrder.insertNewOrders` method assigns to the new orders.
5. Sets initial values of `0.00` for the new order's amounts and charges.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiOrder.NEWORD**

`ccrz__E_Order__c` for the new order, ready to insert.

**SEE ALSO:**

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz\\_\\_E\\_Seller\\_\\_c](#)

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccLogicOrderSplitOrder.setOrderOwner

Assigns an owner to a new order that the `ccrz.ccLogicOrderSplitOrder.copyOrder` method created.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> setOrderOwner (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicOrderSplitOrder.copyOrder` method.

This method requires the following keys in the working map:

**ccrz.ccApiOrder.NEWORD**

`ccrz__E_Order__c` for the new order, which `ccrz.ccLogicOrderSplitOrder.copyOrder` created.

**ccrz.ccApiOrder.PARAM\_OWNERID**

String that specifies the owner for the new order.

**ccrz.ccApiOrder.SPLITID**

String that specifies the criteria to split the original order by, such as seller ID.

**ccrz.ccApiOrder.SRCORD**

`ccrz__E_Order__c`, which specifies the original order record to copy.

### Logic Performed

By default, this method assigns the `ccrz__DefaultRep__c` user from the order's corresponding `ccrz__E_Seller__c` as the new order's owner.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiOrder.NEWORD**

`ccrz__E_Order__c` for the new order, updated with the owner ID and ready to insert.

SEE ALSO:

`ccrz__E_Seller__c`

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccLogicOrderSplitOrder.updateOrders

Executes the DML `update` operation on the original order records that the `ccrz.ccLogicOrderSplitOrder.createNewOrders` method split into child orders.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> updateOrders (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderSplitOrder.createNewOrders` method completes.

This method requires the following key in the working map:

**ccrz.ccApiOrder.ORDERLIST**

`List<ccrz__E_Order__c>` of the original orders to split.

### Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[Inserting and Updating Records](#)

## ccrz.ccLogicOrderSplitOrder.createAdditionalData

Calls the `ccrz.ccApiAddress.createAddress` method to create `ccrz__E_ContactAddr__c` records from the copies of the original order's billing and shipping addresses. To create other data to apply to split orders, override this method in your subscriber extension code.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> createAdditionalData (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderSplitOrder.updateOrders` method completes.

This method requires the following keys in the working map:

#### **ccrz.ccApiOrder.ADDRIDMAP**

`Map<String, Object>`, where:

- Each key is the ID of a `ccrz__E_ContactAddr__c` record for an order's billing or shipping address.
- Each value is a `Map<String, Object>` represents the corresponding `ccrz__E_ContactAddr__c` record.

#### **ccrz.ccApiOrder.OLDADDRDATA**

`List<Map<String, Object>>`, where each `Map<String, Object>` describes a `ccrz__E_ContactAddr__c` record that the original order referenced for a billing or shipping address.

### Logic Performed

This method completes the following actions:

1. Passes `ccrz.ccApiOrder.OLDADDRDATA` as input for the `ccrz.ccApiAddress.createAddress` method. By default, this query applies the following [data size request parameters](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAddress.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

```
}
}
```

2. Calls `ccrz.ccApiAddress.fetch`—because the `ccrz.ccApiAddress.createAddress` input map specified `ccrz.ccApi.SZ_REFETCH => true`—to return the created addresses.
3. Evaluates the `ccrz.ccApiAddress.ADDRESSLIST` output returned from `ccrz.ccApiAddress.fetch`, and creates a map of each original `ccrz__E_ContactAddr__c` record to its new copy.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiOrder.NEWADDRDATA`

`Map<String, List<String>>`, where:

- Each key is the ID of an original `ccrz__E_ContactAddr__c` record.
- Each value is a list of IDs of each created `ccrz__E_ContactAddr__c` record that corresponds to the original `ccrz__E_ContactAddr__c` record.



**Note:** The order of each address list corresponds to the order in which the child orders were split from the parent order.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz.ccApiAddress.createAddress](#)

[ccrz.ccApiAddress.fetch](#)

## `ccrz.ccLogicOrderSplitOrder.insertNewOrders`

Assigns billing and shipping addresses on the new `ccrz__E_Order__c` records, and executes the DML `insert` operation on the `ccrz__E_Order__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> insertNewOrders (Map<String, Object>)
```

## Inputs (Required)


This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderSplitOrder.createAdditionalData` method completes.

This method requires the following keys in the working map:

### `ccrz.ccApiOrder.NEWADDRDATA`

`Map<String, List<String>>`, where:

- Each key is the ID of an original `ccrz__E_ContactAddr__c` record.
- Each value is a list of IDs of each created `ccrz__E_ContactAddr__c` record that corresponds to the original `ccrz__E_ContactAddr__c` record.

 **Note:** The order of each address list corresponds to the order in which the child orders were split from the parent order.

### `ccrz.ccApiOrder.NEWORDLST`

`List<ccrz__E_Order__c>` of the new orders to insert.

## Logic Performed

This method executes the following actions:

1. Loops through the `ccrz.ccApiOrder.NEWORDLST`, and sets the appropriate `ccrz__BillTo__c` and `ccrz__ShipTo__c` addresses.
2. Executes the DML `insert` operation on the new `ccrz__E_Order__c` records.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[Inserting and Updating Records](#)

## `ccrz.ccLogicOrderSplitOrder.createNewOrderItems`

Creates `ccrz__E_OrderItem__c` records for the new orders, based on the specified split order criteria.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createNewOrderItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderSplitOrder.insertNewOrders` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApiOrder.ORDCLNMAP**

`Map<String, Map<String, ccrz__E_Order__c>>`, where:

- The outer map's key is the original order ID.
- Each inner map's key is the criteria for splitting order items: seller ID.
- Each inner map's value is the list of created child `ccrz__E_Order__c` records.

### **ccrz.ccApiOrder.ORDERLIST**

`List<ccrz__E_Order__c>` of the original orders to split.

### **ccrz.ccApiOrder.SPLORDITM**

`Map<String, Map<String, List<ccrz__E_OrderItem__c>>>`, where:

- The outer map's key is the original order ID.
- Each inner map's key is the criteria for splitting order items: seller ID.
- Each inner map's value is the list of corresponding `ccrz__E_OrderItem__c` data.

## Logic Performed

This method completes the following actions:

1. Evaluates the list of created child orders in `ccrz.ccApiOrder.ORDCLNMAP`.
2. Evaluates the list of order items, split by criteria, in `ccrz.ccApiOrder.SPLORDITM`.
3. Calls the `ccrz.ccLogicOrderSplitOrder.copyOrderItem` inner method on each order item, and adds the created `ccrz__E_OrderItem__c` to a list to insert.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiOrder.NEWORDITMLST**

`List<ccrz__E_OrderItem__c>` of the new order items that the `ccrz.ccLogicOrderSplitOrder.copyOrderItem` method creates.

SEE ALSO:

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)



## ccrz.ccLogicOrderSplitOrder.copyOrderItem

Copies an original order item data to a new `ccrz__E_OrderItem__c` record.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> copyOrderItem(Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicOrderSplitOrder.createNewOrderItems` method.

This method requires the following keys in the working map:

**ccrz.ccApiOrder.NEWORD**

`ccrz__E_Order__c`, which specifies the new order record to assign the new order item to.

**ccrz.ccApiOrder.SPLITID**

String that specifies the criteria to split the original order by, such as seller ID.

**ccrz.ccApiOrder.SRCORD**

`ccrz__E_Order__c`, which specifies the original order record that the original order item belonged to.

**ccrz.ccApiOrder.SRCORDITM**

`ccrz__E_OrderItem__c`, which specifies the original order item record to copy.

### Logic Performed

The `ccrz.ccLogicOrderSplitOrder.createNewOrderItems` method calls into this method for creating each new order item. This method completes the following actions for each new order item:

1. Calls the sObject `clone` method to create a duplicate of the original `ccrz__E_OrderItem__c`.
2. Sets the following field values on the new order item:

Field	Value
<code>ccrz__Order__c</code>	The ID of <code>ccrz.ccApiOrder.NEWORD</code>

Field	Value
ccrz__OrderItemId__c	null
ccrz__SplitSource__c	The ID of <code>ccrz.ccApiOrder.SRCORDITM</code> .

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiOrder.NEWORDITM`

`ccrz__E_OrderItem__c` for the new order item, ready to insert.

SEE ALSO:

`ccrz__E_OrderItem__c`

## `ccrz.ccLogicOrderSplitOrder.insertNewOrderItems`

Executes the DML `insert` operation on the new `ccrz__E_OrderItem__c` records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> insertNewOrderItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicOrderSplitOrder.createNewOrderItems` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiOrder.NEWORDITMLST`

`List<ccrz__E_OrderItem__c>` of the new order items to insert.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[Inserting and Updating Records](#)

## ccrz.ccLogicOrderSplitOrder.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the [ccrz.ccLogicOrderSplitOrder.insertNewOrderItems](#) method completes.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApiOrder.ADDRIDMAP`
- `ccrz.ccApiOrder.NEWADDRDATA`
- `ccrz.ccApiOrder.NEWORDITMLST`
- `ccrz.ccApiOrder.NEWORDLST`
- `ccrz.ccApiOrder.OLDADDRDATA`
- `ccrz.ccApiOrder.ORDCLNMAP`
- `ccrz.ccApiOrder.ORDERITEMSBYID`

- `ccrz.ccApiOrder.ORDERLIST`
- `ccrz.ccApiOrder.OWNERPERSPLT`
- `ccrz.ccApiOrder.SPLORDITM`
- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicPCreate

Define the business logic for creating product index records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiProductIndex.create](#)

### Methods

#### IN THIS SECTION:

##### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

##### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

##### [fetchLocaleData](#)

Queries data for specific locales, if specified, or all locales for the storefront.

##### [fetchPriceListItems](#)

Calls the `ccrz.ccApiPriceList.fetchPriceListItems` method for returning price list items to create product index records for. This method returns data only when the original input data doesn't already specify

`ccrz.ccApiProductIndex.PRICE_LIST_ITEMS`.

##### [processPriceListItems](#)

Organizes price list by items by product ID and currency.

[fetchProductIndices](#)

Calls the `ccrz.ccApiProductIndex.fetch` method for returning existing product index records for the products that corresponding to processed price list items.

[updateProductIndices](#)

Updates any existing product index records that previous iterations of the batch job created and ensures that each record's index price is set.

[buildIndex](#)

Iterates over each locale and calls inner methods for returning product records and creating product index records.

[fetchProducts](#)

Begins the For loop within the `ccrz.ccLogicPICreate.buildIndex` method by returning product records for a specific locale.

[createProductIndices](#)

Concludes the For loop within the `ccrz.ccLogicPICreate.buildIndex` method by creating product index records for the fetched products.

[upsertProductIndices](#)

Executes the DML `upsert` operation for the created product index records, which assigns IDs to each record. This operation updates existing product index records with new price and price list information, and inserts new records that this logic provider chain created.

[refetch](#)

Calls the `ccrz.ccApiProductIndex.fetch` method for fetching the created product index records.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## SEE ALSO:

[Salesforce Help Maintain Product Index Records](#)

## ccrz.ccLogicPICreate.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiProductIndex.create` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicPICreate` class in the following order:

1. `processInput`
2. `fetchLocaleData`
3. `fetchPriceListItems`
4. `fetchProductIndices`
5. `updateProductIndices`
6. `buildIndex`, which executes a For loop that calls into the following inner methods for each locale:
  - a. `fetchProducts`
  - b. `createProductIndices`
7. `upsertProductIndices`
8. `refetch`
9. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.



**Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicPICreate.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPICreate.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

### `ccrz.ccApiProductIndex.PRICE_LIST_ITEM_IDS`

`Set<String>` of Salesforce IDs of [price list item](#) records that you want to create product index records for.

```
new Set<String>{ 'Price_List_Item_ID_1', 'Price_List_Item_ID_2' }
```

### `ccrz.ccApiProductIndex.PRICE_LIST_ITEMS`

`Map<String, Object>` of price list item records, keyed by ID, that you want to create product index records for.

### `ccrz.ccApiProductIndex.PRODUCT_IDS`

`Set<String>` of Salesforce IDs of [product](#) records that you want to create product index records for.

```
new Set<String>{ 'Product_ID_1', 'Product_ID_2' }
```

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccLogic.ORIGINAL_INPUT`

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicPICreate.prepReturn` method removes this intermediate key from the working map.

SEE ALSO:

[ccrz\\_\\_E\\_PricelistItem\\_\\_c](#)

[ccrz\\_\\_E\\_Product\\_\\_c](#)

[Salesforce Help Maintain Product Index Records](#)

## `ccrz.ccLogicPICreate.fetchLocaleData`

Queries data for specific locales, if specified, or all locales for the storefront.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchLocaleData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPICreate.processInput` method completes.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiProductIndex.LOCALES`

`Set<String>` of locales that you want to create product index records for.

```
new Set<String>{'en_US', 'ja_JP'}
```

 **Note:** If the input data doesn't specify this key, this method queries all locales.

## Logic Performed

This method evaluates whether the input data includes `ccrz.ccApiProductIndex.LOCALES`:

- If the input data includes `ccrz.ccApiProductIndex.LOCALES`, this method returns those locales in the output data.
- If the input data doesn't include `ccrz.ccApiProductIndex.LOCALES`, this method queries the storefront settings object, `ccrz__E_StorefrontSettings__c`, for its specified locales. This return data includes the storefront's enabled full locales (such as `en_US` and `ja_JP`) and language-only locales (such as `en` or `ja`).

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiProductIndex.LOCALES`

`Set<String>` of specific locales, if specified, or all locales.

SEE ALSO:

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)



## ccrz.ccLogicPCreate.fetchPriceListItems

Calls the `ccrz.ccApiPriceList.fetchPriceListItems` method for returning price list items to create product index records for. This method returns data only when the original input data doesn't already specify `ccrz.ccApiProductIndex.PRICE_LIST_ITEMS`.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchPriceListItems (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPCreate.fetchLocaleData` method completes.

### Logic Performed

If the input data already specified `ccrz.ccApiProductIndex.PRICE_LIST_ITEMS`, this method returns the input data, and execution continues to the next method in the logic chain.

If the input data didn't specify `ccrz.ccApiProductIndex.PRICE_LIST_ITEMS`, this method completes the following actions:

1. Calls `ccrz.ccApiPriceList.fetchPriceListItems` using the following inputs, depending on which key was specified in the original input data:

Logic Chain Input	ccrz.ccApiPriceList.fetchPriceListItems Input
<code>ccrz.ccApiProductIndex.PRICE_LIST_ITEM_IDS</code>	<code>ccrz.ccApiPriceList.PLI_IDS</code>
<code>ccrz.ccApiProductIndex.PRODUCT_IDS</code>	<code>ccrz.ccApiPriceList.PRODUCTIDS</code>

2. Parses the `ccrz.ccApiPriceList.PRICELISTITEMS` output from `ccrz.ccApiPriceList.fetchPriceListItems`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiProductIndex.PRICE_LIST_ITEMS`

`Map<String, Object>` of `ccrz__E_PriceListItem__c` records that the `ccrz.ccApiPriceList.fetchPriceListItems` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_PricelistItem\\_\\_c](#)

[ccrz.ccApiPriceList.fetchPriceListItems](#)

## `ccrz.ccLogicPICreate.processPriceListItems`

Organizes price list by items by product ID and currency.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processPriceListItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPICreate.fetchPriceListItems` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiProductIndex.PRICE_LIST_ITEMS`

`Map<String, Object>` of `ccrz__E_PriceListItem__c` records.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiProductIndex.MASTER_PRICE_LIST`

String that specifies the ID of the storefront's [master price list](#), if applicable.

## Logic Performed

If the input data specified `ccrz.ccApiProductIndex.MASTER_PRICE_LIST`, this method calls the private inner method, `processPriceListItemsMasterPriceList`, which completes the following actions:

If the input data didn't specify `ccrz.ccApiProductIndex.MASTER_PRICE_LIST`, this method calls the private inner method, `processPriceListItemsWithoutMasterPriceList`.

Each inner method completes the following actions:

1. Parses `ccrz.ccApiProductIndex.PRICE_LIST_ITEMS`.
2. Sets an indexed price for each product:
  - When `ccrz.ccApiProductIndex.MASTER_PRICE_LIST` is specified, the indexed price is the product's price in the master price list.
  - When `ccrz.ccApiProductIndex.MASTER_PRICE_LIST` isn't specified, the indexed price is the product's *lowest* price from all the price list items assigned to the product in that currency.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiProductIndex.CURRENCIES**

`Set<String>` of currency codes for the price list items.



**Note:** When `ccrz.ccApiProductIndex.MASTER_PRICE_LIST` is specified, this key's value is 'GLOBAL'.

### **ccrz.ccApiProductIndex.PRODUCT\_PRICING\_DATA**

`Map<String, Map<String, Map<String, Object>>>` of pricing data for the price list items, where:

- The outer map is keyed by product ID
- The second-level map is keyed by currency ISO code
- The inner map describes the indexed price and its corresponding price list

SEE ALSO:

[ccrz\\_\\_E\\_PricelistItem\\_\\_c](#)

[Salesforce Help Strategies for Managing Price List Item Currencies with Product Index Records](#)

## ccrz.ccLogicPICreate.fetchProductIndices

Calls the `ccrz.ccApiProductIndex.fetch` method for returning existing product index records for the products that corresponding to processed price list items.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchProductIndices (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPICreate.processPriceListItems` method completes.

This method requires the following keys in the working map:

### `ccrz.ccApiProductIndex.CURRENCIES`

`Set<String>` of currency codes for the price list items.

### `ccrz.ccApiProductIndex.PRODUCT_PRICING_DATA`

`Map<String, Map<String, Map<String, Object>>>` of pricing data for the price list items, where:

- The outer map is keyed by product ID
- The second-level map is keyed by currency ISO code
- The inner map describes the indexed price and its corresponding price list

## Logic Performed

This method passes the following inputs to `ccrz.ccApiProductIndex.fetch`

Logic Chain Input	<code>ccrz.ccApiProductIndex.fetch</code> Input
<code>ccrz.ccApiProductIndex.PRODUCT_PRICING_DATA</code> using the <code>keySet</code> method of the Apex <code>Map</code> class	<code>ccrz.ccApiProductIndex.PRODUCT_IDS</code>
<code>ccrz.ccApiProductIndex.CURRENCIES</code>	<code>ccrz.ccApiProductIndex.CURRENCIES</code>

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiProductIndex.PRODUCT_INDICES`

`List<ccrz__E_ProductIndex__c>` that the `ccrz.ccApiProductIndex.fetch` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

[ccrz.ccApiProductIndex.fetch](#)

[keySet \(\)](#)

## ccrz.ccLogicPICreate.updateProductIndices

Updates any existing product index records that previous iterations of the batch job created and ensures that each record's index price is set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> updateProductIndices (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPICreate.fetchProductIndices` method completes.

This method requires the following keys in the working map:

#### **ccrz.ccApiProductIndex.PRODUCT\_INDICES**

`List<ccrz__E_ProductIndex__c>` of existing product index records that match the products included in `ccrz.ccApiProductIndex.PRODUCT_PRICING_DATA`.

#### **ccrz.ccApiProductIndex.PRODUCT\_PRICING\_DATA**

`Map<String, Map<String, Map<String, Object>>>` of pricing data for the price list items, where:


- The outer map is keyed by product ID
- The second-level map is keyed by currency ISO code
- The inner map describes the indexed price and its corresponding price list

### Logic Performed

For each `ccrz__E_ProductIndex__c` record in `ccrz.ccApiProductIndex.PRODUCT_INDICES`, this method completes the following actions:

1. Evaluates whether the product's current price from `ccrz.ccApiProductIndex.PRODUCT_PRICING_DATA` is less than the existing `ccrz__IndexPrice__c` on the product index record, and sets `ccrz__IndexPrice__c` to that current price.

- Checks the length of the `ccrz__PriceLists__c` field on a product index record where `ccrz__IndexType__c` is `SEARCH`. When the length exceeds the allowed field size, this method creates another product index record for the remaining price lists.

 **Note:** This method doesn't evaluate the length of the `ccrz__PriceLists__c` field for any other `ccrz__IndexType__c` value.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**`ccrz.ccApiProductIndex.PRODUCT_INDICES`**

`List<ccrz__E_ProductIndex__c>` of updated product index records, ready to upsert.

SEE ALSO:

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

[keySet \(\)](#)

[Salesforce Help Strategies for Managing Price List Item Currencies with Product Index Records](#)

## ccrz.ccLogicPICreate.buildIndex

Iterates over each locale and calls inner methods for returning product records and creating product index records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> buildIndex (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPICreate.updateProductIndices` method completes.

This method requires the following key in the working map:

**ccrz.ccApiProductIndex.LOCALES**

Set<String> of locales that the `ccrz.ccLogicPCreate.fetchLocaleData` method returned.

## Logic Performed

For each locale in `ccrz.ccApiProductIndex.LOCALES`, this method completes the following actions:

1. Instantiates an input key, `ccrz.ccApiProductIndex.LOCALE`, for the current locale and calls the `ccrz.ccLogicPCreate.fetchProducts` inner method for returning product records for the locale.
2. Calls the `ccrz.ccLogicPCreate.createProductIndices` inner method for creating product index records for the fetched products.

## Outputs

This method returns the working `Map<String, Object>` after the inner loop methods complete.

## ccrz.ccLogicPCreate.fetchProducts

Begins the For loop within the `ccrz.ccLogicPCreate.buildIndex` method by returning product records for a specific locale.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchProducts (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPCreate.buildIndex` method.

This method requires the following keys in the working map:

**ccrz.ccApiProductIndex.LOCALE**

String that specifies the current locale evaluated in the `ccrz.ccLogicPCreate.buildIndex` For loop.

**ccrz.ccApiProductIndex.PRODUCT\_PRICING\_DATA**

`Map<String, Map<String, Map<String, Object>>>` of pricing data for the price list items, where:

- The outer map is keyed by product ID
- The second-level map is keyed by currency ISO code
- The inner map describes the indexed price and its corresponding price list

## Logic Performed

This method passes the following inputs to `ccrz.ccApiProduct.fetch`:

Logic Chain Input	ccrz.ccApiProduct.fetch Input
<code>ccrz.ccApiProductIndex.PRODUCT_PRICING_DATA</code> using the <a href="#">keySet</a> method of the Apex Map class	<code>ccrz.ccApiProduct.PRODUCTIDLIST</code>
<code>ccrz.ccApiProductIndex.LOCALE</code>	<code>ccrz.ccApiProduct.PRODUCTLOCALE</code>

By default, this query applies the following [data size request parameters](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProduct.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_M,
        ccrz.ccApi.SZ_ASSC => false,
        ccrz.ccApi.SZ_REL => new List<String>{'Product_Categories__r',
'XLProduct_Item_I18Ns__r'}
    }
}
```

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiProduct.PRODUCTLIST`

`List<Object>` of `ccrz__E_Product__c` records that the `ccrz.ccApiProduct.fetch` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_Product\\_\\_c](#)  
[ccrz.ccApiProduct.fetch](#)

## ccrz.ccLogicPICreate.createProductIndices

Concludes the For loop within the `ccrz.ccLogicPICreate.buildIndex` method by creating product index records for the fetched products.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createProductIndices (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPICreate.buildIndex` method.

This method requires the following keys in the working map:

### **ccrz.ccApiProduct.PRODUCTLIST**

List<Object> of `ccrz__E_Product__c` records that the `ccrz.ccApiProduct.fetch` method returned during `ccrz.ccLogicPICreate.fetchProducts`.

### **ccrz.ccApiProductIndex.PRODUCT\_PRICING\_DATA**

Map<String, Map<String, Map<String, Object>>> of pricing data for the price list items, where:

- The outer map is keyed by product ID
- The second-level map is keyed by currency ISO code
- The inner map describes the indexed price and its corresponding price list

## Logic Performed

For each product in `ccrz.ccApiProduct.PRODUCTLIST`, this method completes the following actions:

1. Finds the corresponding product ID in `ccrz.ccApiProductIndex.PRODUCT_PRICING_DATA`.
2. Evaluates whether the product has any related `ccrz__E_ProductCategory__c` assignments, and creates a product index record each category assignment. These product index records provide results on the Product List page when a buyer selects a specific category.
3. Creates a product index record for the product's result on the Product List page when a buyer searches the storefront by keyword.
4. Adds the created product index records to `ccrz.ccApiProductIndex.PRODUCT_INDICES`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiProductIndex.PRODUCT\_INDICES**

List<`ccrz__E_ProductIndex__c`> of updated product index records, ready to upsert.

SEE ALSO:

[ccrz\\_\\_E\\_ProductCategory\\_\\_c](#)

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

[Salesforce Help Product List Page](#)

## ccrz.ccLogicPCreate.upsertProductIndices

Executes the DML `upsert` operation for the created product index records, which assigns IDs to each record. This operation updates existing product index records with new price and price list information, and inserts new records that this logic provider chain created.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> upsertProductIndices (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPCreate.buildIndex` method completes.

This method requires the following key in the working map:

**ccrz.ccApiProductIndex.PRODUCT\_INDICES**

`List<ccrz__E_ProductIndex__c>` of product index records to upsert.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiProductIndex.IDS**

`Set<String>` of the upserted product index records.

SEE ALSO:

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

[Upserting Records](#)

## ccrz.ccLogicPCreate.refetch

Calls the `ccrz.ccApiProductIndex.fetch` method for fetching the created product index records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPICreate.upsertProductIndices` method completes.

This method requires the following keys in the working map:

### `ccrz.ccApi.SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProductIndex.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

### `ccrz.ccApiProductIndex.IDS`

`Set<String>` of the upserted product index records.

## Outputs

This method returns the working `Map<String, Object>`, which includes the `ccrz.ccApiProductIndex.fetch` outputs.

SEE ALSO:

[ccrz\\_\\_E\\_ProductIndex\\_\\_c](#)

[ccrz.ccApiProductIndex.fetch](#)

## ccrz.ccLogicPICreate.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPICreate.refetch` method completes.

### Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApiProduct.PRODUCTLIST`
- `ccrz.ccApiProductIndex.CURRENCIES`
- `ccrz.ccApiProductIndex.LOCALE`
- `ccrz.ccApiProductIndex.LOCALES`
- `ccrz.ccApiProductIndex.PRICE_LIST_ITEMS`
- `ccrz.ccApiProductIndex.PRICE_LIST_ITEM_IDS`
- `ccrz.ccApiProductIndex.PRODUCT_PRICING_DATA`
- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicPIFetchEntitled

Define the business logic for returning a buyer's entitled product IDs that correspond to product index records queried from a search string or specified categories.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiProductIndex.fetchEntitled](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [fetchSpecFilters](#)

Evaluate any specs and spec values included in the input map for an extra layer of filtering entitled products. This method invokes the `ccrz.ccApiSpec.fetchProductSpecIndices` method to query product spec index records and return a map of specs and entitled products that have values for those specs.

#### [fetchEntitledPriceLists](#)

Call the `ccrz.ccApiPriceList.fetchEntitled` method for returning the buyer's entitled price lists, which filter the product index query for only those records that reference those price lists.

#### [performIndexSearch](#)

Call the `ccrz.ccApiProductIndex.search` method for performing a SOSL search of product index records that match a search string specified in the working map.

#### [performIndexFetch](#)

Call the `ccrz.ccApiProductIndex.fetch` method for performing a SOQL query of product index records that match category IDs specified in the working map.

#### [processProductIndices](#)

Determine how many entitled product index records exist that match the product index query. Evaluate whether subsequent calls are necessary for returning the full set of results, and identify the last product index record in the current set of results.

[rankSearchResultsAndProcess](#)

Rank the products that correspond to product index records returned from a search string query. This method determines how search results sort on the Product List page when a buyer sorts by relevance. This method determines relevance by evaluating the product's name, short description, and SKU for occurrences of the search string.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicPIFetchEntitled.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiProductIndex.fetchEntitled` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicPIFetchEntitled` class in the following order:

1. `processInput`
2. `fetchSpecFilters`
3. `fetchEntitledPriceLists`
4. `performIndexSearch` or `performIndexFetch`
5. `processProductIndices`, which can call into `rankSearchResultsAndProcess`
6. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicPIFetchEntitled.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicPIFetchEntitled.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

### **ccrz.ccApiProductIndex.CATEGORY\_IDS**

`Set<String>` of specific category IDs passed to the `ccrz.ccApiProductIndex.fetch` method, which executes a SOQL query of product index records that reference the specified category IDs.

### **ccrz.ccService.SEARCHSTRING**

String passed to the `ccrz.ccApiProductIndex.search` method for executing a SOSL query of product index records.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicPIFetchEntitled.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicPIFetchEntitled.fetchSpecFilters

Evaluate any specs and spec values included in the input map for an extra layer of filtering entitled products. This method invokes the `ccrz.ccApiSpec.fetchProductSpecIndices` method to query product spec index records and return a map of specs and entitled products that have values for those specs.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchSpecFilters(Map<String, Object>)
```

### Inputs (Required)


This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPIFetchEntitled.processInput` method.

This method requires the following key in the working map:

#### **ccrz.ccApiProductIndex.PRODUCT\_SPEC\_FILTERS**

`List<Map<String, Object>>` or `List<Object>` of specs and spec values to pass to the `ccrz.ccApiSpec.PRODUCT_SPEC_INDICES` key for the `ccrz.ccApiSpec.fetchProductSpecIndices` method. Include each spec's ID and the values that you want to filter by, such as:


```
[ {
  "specValues" : [ {
    "value" : "Blue"
  } ],
  "sfid" : "alh6A00000045s5QAA"
} ]
```

 **Note:** If you specify multiple values for the same spec, the method applies an OR condition for evaluating all values.

For a spec where Filter Type is Slider, you can specify a `filterMin` and `filterMax` range for filtering:

```
[ {
  "sfid" : "alh6A00000045s5QAA",
  "filterMin" : "1.0",
  "filterMax" : "10.0"
} ]
```



 **Note:** This method checks the value of the Spec Value Query Limit setting of the Product List storefront configuration module. By default, the method queries at most 20,000 spec values. You can override this default value with a custom limit. The complexity of your product data can also affect the scope of the query, regardless of a limit that you specify.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiProductIndex.PRODUCT_FILTERS`

`Map<String, List<String>>` that relates each matching spec ID with a list of entitled product IDs that have spec values for that spec.

SEE ALSO:

[ccrz.ccApiSpec.fetchProductSpecIndices](#)

## `ccrz.ccLogicPIFetchEntitled.fetchEntitledPricelists`

Call the `ccrz.ccApiPriceList.fetchEntitled` method for returning the buyer's entitled price lists, which filter the product index query for only those records that reference those price lists.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchEntitledPriceLists (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the [ccrz.ccLogicPIFetchEntitled.fetchSpecFilters](#) method.

This method executes only if the working map doesn't already specify the following key:

### `ccrz.ccApiProductIndex.PRICE_LISTS`

`Set<String>` of price list IDs, which filters the product index query for only those records that reference the specified price lists. If you exclude this key from the input map, the method invokes the [ccrz.ccApiPriceList.fetchEntitled](#) method and filters the query by all the user's entitled price lists.

## Outputs

This method returns the working `Map<String, Object>`, and adds the following intermediate key:

**ccrz.ccApiProductIndex.PRICE\_LISTS**  
`Set<String>` of price list IDs that the buyer is entitled to.

The `ccrz.ccLogicPIFetchEntitled.prepReturn` method removes this intermediate key from the working map.

SEE ALSO:

[ccrz.ccApiPriceList.fetchEntitled](#)

## ccrz.ccLogicPIFetchEntitled.performIndexSearch

Call the `ccrz.ccApiProductIndex.search` method for performing a SOSL search of product index records that match a search string specified in the working map.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> performIndexSearch (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPIFetchEntitled.fetchEntitledPriceLists` method.


This method executes only if the working map includes both of the following keys:

**ccrz.ccService.SEARCHSTRING**

String passed to the `ccrz.ccApiProductIndex.search` method for executing a SOSL query of product index records.

**ccrz.ccApiProductIndex.PRICE\_LISTS**

`Set<String>` of price list IDs, which filters the product index query for only those records that reference the specified price lists. If you exclude this key from the input map, the method invokes the `ccrz.ccApiPriceList.fetchEntitled` method and filters the query by all the user's entitled price lists.

 **Note:** If the working map includes `ccrz.ccApiProductIndex.CATEGORY_IDS` instead of `ccrz.ccService.SEARCHSTRING`, this method is skipped and execution continues to the `ccrz.ccLogicPIFetchEntitled.performIndexFetch` method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiProductIndex.PRODUCT_INDICES`

`List<ccrz__E_ProductIndex__c>` of entitled product index records that match the search string.

SEE ALSO:

[ccrz.ccApiProductIndex.search](#)

## `ccrz.ccLogicPIFetchEntitled.performIndexFetch`

Call the `ccrz.ccApiProductIndex.fetch` method for performing a SOQL query of product index records that match category IDs specified in the working map.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> performIndexFetch(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPIFetchEntitled.performIndexSearch` method.

This method executes only if the working map includes both of the following keys:

### `ccrz.ccApiProductIndex.CATEGORY_IDS`

`Set<String>` of specific category IDs passed to the `ccrz.ccApiProductIndex.fetch` method, which executes a SOQL query of product index records that reference the specified category IDs.

**ccrz.ccApiProductIndex.PRICE\_LISTS**

Set<String> of price list IDs, which filters the product index query for only those records that reference the specified price lists. If you exclude this key from the input map, the method invokes the [ccrz.ccApiPriceList.fetchEntitled](#) method and filters the query by all the user's entitled price lists.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApiProductIndex.PRODUCT\_INDICES**

List<ccrz\_\_E\_ProductIndex\_\_c> of entitled product index records that match the specified category IDs.

SEE ALSO:

[ccrz.ccApiProductIndex.fetch](#)

## ccrz.ccLogicPIFetchEntitled.processProductIndices

Determine how many entitled product index records exist that match the product index query. Evaluate whether subsequent calls are necessary for returning the full set of results, and identify the last product index record in the current set of results.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processProductIndices (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working Map<String, Object> from the [ccrz.ccLogicPIFetchEntitled.performIndexSearch](#) or [ccrz.ccLogicPIFetchEntitled.performIndexFetch](#) method.

This method requires the following key in the working map:

**ccrz.ccApiProductIndex.PRODUCT\_INDICES**

List<ccrz\_\_E\_ProductIndex\_\_c> of entitled product index records that match either a search string or specified category IDs.

## Inputs (Optional)

This method also evaluates the following optional keys, which are included in the working map if specified in the [ccrz.ccApiProductIndex.fetchEntitled](#) method's input map.

### **ccrz.ccApiProductIndex.INCLUDE\_COUNT**

Boolean

Value	Usage
<i>true</i>	Include the <code>ccrz.ccApiProductIndex.COUNT</code> key in the return data, which indicates the total number of product index records that match the query.
<i>false</i> (default)	Don't include the <code>ccrz.ccApiProductIndex.COUNT</code> key in the return data.

### **ccrz.ccApiProductIndex.INDEX\_LIMIT**

Integer that specifies a maximum number of product index records to query, which affects how many product IDs are returned.

### **ccrz.ccApiProductIndex.PRODUCT\_IDS\_ONLY**


Boolean



Value	Usage
<i>true</i>	Return only <code>ccrz.ccApiProductIndex.PRODUCT_IDS</code> , and skip calculating <code>ccrz.ccApiProductIndex.CURRENT_INDEX</code> .
<i>false</i> (default)	Calculate <code>ccrz.ccApiProductIndex.CURRENT_INDEX</code> for the return data.

### **ccrz.ccService.ORDERBY**

String that specifies a key, which indicates how you want to sort product index records. For example:

```
ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_NAME
```

Value	Usage
<code>ccrz.ccApiProductIndex.BY_NAME</code>	Sort by the localized name of the product that the product index record references.
<code>ccrz.ccApiProductIndex.BY_PRICE</code>	Sort by the Index Price field on the product index record.
<code>ccrz.ccApiProductIndex.BY_RELEVANCE</code>	Sort by occurrences of the search string in the product's name, short description, and SKU.   <b>Note:</b> <ul style="list-style-type: none"> <li>Relevance sorting is available only for product index records that match a search string.</li> <li>When this key is specified, this method calls into the <a href="#">ccrz.ccLogicPIFetchEntitled.rankSearchResultsAndProcess</a> method for executing the relevance sorting.</li> </ul>

Value	Usage
<code>ccrz.ccApiProductIndex.BY_SEQUENCE</code>	Sort by the Sequence field on the <a href="#">product category record</a> that the product index record references.   <b>Note:</b> <ul style="list-style-type: none"> <li>Sequence sorting is available only for product index records that match specified category IDs.</li> <li>B2B Commerce for Visualforce Summer '19 (managed package version 4.11, API version 10) or later flattens the data model required for referencing this Sequence value. The product index record includes a Product Category Sequence field, whose value is copied from the corresponding product category record's Sequence field.</li> </ul>
<code>ccrz.ccApiProductIndex.BY_START_DATE</code>	Sort by the value of the Start Date field on the product that the product index record references.   <b>Note:</b> B2B Commerce for Visualforce Summer '19 (managed package version 4.11, API version 10) or later flattens the data model required for referencing this Start Date value. The product index record includes a Product Start Date field, whose value is copied from the corresponding product record's Start Date field.

## Outputs

This method returns the working `Map<String, Object>`, and adds the following keys:

### `ccrz.ccApiProductIndex.CURRENT_INDEX`

`Map<String, Object>` that specifies the last product index record within the current set of sorted results. You can pass this map to subsequent `ccrz.ccApiProductIndex.fetch`, `ccrz.ccApiProductIndex.search`, or `ccrz.ccApiProductIndex.fetchEntitled` method calls for querying the next set of results. This map specifies the following subkeys:

Key	Usage
<code>primary</code>	The value that the <code>ccrz.ccService.ORDERBY</code> input key used for sorting product index records.
<code>secondary</code>	The ID of the referenced product index record.

For example:

```
{
  "primary": "sort value, such as index price or product name",
  "secondary": "Product_Index_ID"
}
```

### `ccrz.ccApiProductIndex.HAS_MORE`

Boolean

Value	Usage
<code>true</code>	The method returned more results than the limit that the <code>ccrz.ccApiProductIndex.INDEX_LIMIT</code> input key specified.
<code>false</code>	The method returned fewer results than the limit that the <code>ccrz.ccApiProductIndex.INDEX_LIMIT</code> input key specified.

**ccrz.ccApiProductIndex.PRODUCT\_CATEGORY\_SEQUENCE**

Map<String, Decimal> that relates each entitled product ID to its corresponding product category sequence value. This key is returned only when the input data includes:

```
ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_SEQUENCE
```

**ccrz.ccApiProductIndex.PRODUCT\_IDS**

List<String> of entitled product IDs from the queried product index records. The number of product IDs included in this list depends on whether the input data specified a `ccrz.ccApiProductIndex.INDEX_LIMIT` value.

SEE ALSO:

[Salesforce Help Factors That Affect Sorting by Price on the Product List Page](#)

**ccrz.ccLogicPIFetchEntitled.rankSearchResultsAndProcess**

Rank the products that correspond to product index records returned from a search string query. This method determines how search results sort on the Product List page when a buyer sorts by relevance. This method determines relevance by evaluating the product's name, short description, and SKU for occurrences of the search string.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> rankSearchResultsAndProcess (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPIFetchEntitled.processProductIndices` method.

This method executes only if the working map includes the following keys:

### `ccrz.ccApiProductIndex.PRODUCT_INDICES`

`List<ccrz__E_ProductIndex__c>` of entitled product index records that match a search string. The `ccrz.ccLogicPIFetchEntitled.performIndexSearch` method returns this key.

### `ccrz.ccService.ORDERBY`

String that specifies that search results sort by relevance:

```
ccrz.ccService.ORDERBY => ccrz.ccApiProductIndex.BY_RELEVANCE
```

If this key specifies a different sorting method, the method doesn't execute.

### `ccrz.ccService.SEARCHSTRING`

String passed to the `ccrz.ccApiProductIndex.search` method for executing a SOSL query of product index records.

## Inputs (Optional)

This method also evaluates the following optional keys, which are included in the working map if specified in the `ccrz.ccApiProductIndex.fetchEntitled` method's input map.

### `ccrz.ccApiProductIndex.INCLUDE_COUNT`

Boolean

Value	Usage
<code>true</code>	Include the <code>ccrz.ccApiProductIndex.COUNT</code> key in the return data, which indicates the total number of product index records that match the query.
<code>false</code> (default)	Don't include the <code>ccrz.ccApiProductIndex.COUNT</code> key in the return data.

### `ccrz.ccApiProductIndex.INDEX_LIMIT`

Integer that specifies a maximum number of product index records to query, which affects how many product IDs are returned.

## Outputs

This method returns the working `Map<String, Object>`, and adds the following keys:

### `ccrz.ccApiProductIndex.CURRENT_INDEX`

`Map<String, Object>` that specifies the last product index record within the current set of sorted results. You can pass this map to subsequent method calls for querying the next set of results. This map specifies the following subkeys:

Key	Usage
<code>primary</code>	The rank determined for the product index record.
<code>secondary</code>	The ID of the referenced product index record.



For example:

```
{
  "primary": "250",
  "secondary": "an18digitProductID"
}
```

#### **ccrz.ccApiProductIndex.HAS\_MORE**

Boolean

Value	Usage
<i>true</i>	The method returned more results than the limit that the <code>ccrz.ccApiProductIndex.INDEX_LIMIT</code> input key specified.
<i>false</i>	The method returned fewer results than the limit that the <code>ccrz.ccApiProductIndex.INDEX_LIMIT</code> input key specified.

#### **ccrz.ccApiProductIndex.PRODUCT\_IDS**

`List<String>` of entitled product IDs from the queried product index records. The number of product IDs included in this list depends on whether the input data specified a `ccrz.ccApiProductIndex.INDEX_LIMIT` value.

#### **ccrz.ccApiProductIndex.PRODUCT\_SEARCH\_RANKINGS**

`Map<String, Integer>` that relates each entitled product ID to its determined relevance rank. For example:

```
{
  "an18digitProductID" : 100,
  "an18digitProductID" : 200,
  "an18digitProductID" : 250,
  "an18digitProductID" : 50
}
```

## ccrz.ccLogicPIFetchEntitled.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPIFetchEntitled.processProductIndices` method.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccApiProductIndex.PRICE_LISTS`
- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicPIRemove

Define the business logic for deleting product index records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiProductIndex.remove](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

[fetchProductIndices](#)

Calls the `ccrz.ccApiProductIndex.fetch` method for returning the product index records that you want to delete.

[deleteProductIndices](#)

Executes the DML `delete` operation for the product index records.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## SEE ALSO:

[Salesforce Help Maintain Product Index Records](#)

## ccrz.ccLogicPIRemove.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiProductIndex.remove` method.

### Logic Performed


This method invokes other methods of the `ccrz.ccLogicPIRemove` class in the following order:

1. `processInput`
2. `fetchProductIndices`
3. `deleteProductIndices`

#### 4. prepReturn

### Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicPIRemove.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPIRemove.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

#### **ccrz.ccApiProductIndex.IDS**

`Set<String>` of Salesforce IDs of product index records to delete.


```
new Set<String>{'Product_Index_ID_1', 'Product_Index_ID_2'}
```

### Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

#### **ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.  <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint.</a>

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The [ccrz.ccLogicPIRemove.prepReturn](#) method removes this intermediate key from the working map.

**ccrz.ccLogicPIRemove.fetchProductIndices**

Calls the [ccrz.ccApiProductIndex.fetch](#) method for returning the product index records that you want to delete.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> fetchProductIndices (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> after the [ccrz.ccLogicPIRemove.processInput](#) method completes.

This method requires the following key in the working map:

**ccrz.ccApiProductIndex.IDS**

Set<String> of Salesforce IDs of the [product index records](#) to delete.

## Logic Performed

This method passes `ccrz.ccApiProductIndex.IDS` to `ccrz.ccApiProductIndex.fetch`. By default, this query applies the following [data size request parameters](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiProductIndex.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_S,
        ccrz.ccApi.SZ_ASSC => false
        ccrz.ccApi.SZ_SKIPTRZ => true
    }
}
```

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiProductIndex.PRODUCT\_INDICES**

`List<ccrz__E_ProductIndex__c>` of the product index records to delete. The `ccrz.ccApiProductIndex.fetch` method returned this list.

SEE ALSO:

[ccrz.ccApiProductIndex.fetch](#)

## ccrz.ccLogicPIRemove.deleteProductIndices

Executes the DML `delete` operation for the product index records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> deleteProductIndices (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicPIRemove.fetchProductIndices` method completes.

This method requires the following key in the working map:

**ccrz.ccApiProductIndex.PRODUCT\_INDICES**

List<ccrz\_\_E\_ProductIndex\_\_c> of the product index records to delete.

## Outputs

This method returns the working Map<String, Object>, and doesn't add or remove any keys.

SEE ALSO:

[Deleting Records](#)

## ccrz.ccLogicPIRemove.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working Map<String, Object> after the `ccrz.ccLogicPIRemove.deleteProductIndices` method completes.

## Outputs

This method removes the following intermediate key from the working map, and returns the final Map<String, Object>.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicPLEntLists

Define the business logic for returning an account's entitled price lists.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiPriceList.fetchEntitled](#)

## Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

[createPricingContext](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

[fetchPriceGroupPriceListIDs](#)

Returns the IDs of price lists associated with the account's price group, if applicable.

[fetchAccountGroupPriceListIDs](#)

Returns the IDs of price lists associated with the account's account group, if applicable. This method invokes the `ccrz.ccServiceDAO.doQuery` method for executing the query.

[performRefetch](#)

Call the `ccrz.ccApiPriceList.fetch` method for fetching the price lists and price list items that correspond to the price list IDs returned from the previous methods.

## ccrz.ccLogicPLEntLists.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiPriceList.fetchEntitled` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicPLEntLists` class in the following order:

1. `createPricingContext`
2. `fetchPriceGroupPriceListIDs`
3. `fetchAccountGroupPriceListIDs`
4. `performRefetch`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.



**Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicPLEntLists.createPricingContext

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createPricingContext (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicPLEntLists.process` method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key values if they're not specified in the input data.

### **ccrz.ccApiPriceList.ALLWMIX**

Boolean

Value	Usage
<i>true</i>	Allow mixed pricing queries when you filter by price group. The method tries to match the specified price group when returning price list records. If a price group match isn't available, the method falls back to the specified account's account group.
<i>false</i> (default)	Don't allow mixed pricing queries when you filter by price group. The method only tries to match the specified price group when returning price list records.

If this key isn't specified in the input data, the method falls back to the value of the `ccrz__E_AccountSettings__c.ccrz__AllowMixedPricingCarts__c` field.

### **ccrz.ccApiPriceList.CURRCODE**

String that specifies an ISO 4217 currency code, such as *USD* or *JPY*, to filter entitled price lists by. If this key isn't specified in the input data, the method falls back to the value of the `ccrz.cc_CallContext.userCurrency` field.

### **ccrz.ccApiPriceList.EFFACCTID**

String that specifies an [account ID](#) record to filter entitled price lists by. The method returns only the price list records that are assigned to the specified account's account group. If this key isn't specified in the input data, the method falls back to the value of the `ccrz.cc_CallContext.currAccountId` field.

### **ccrz.ccApiPriceList.EFFDATE**

Date to filter price lists by. The method returns only the price list records where the date is included in the Start Date and End Date range. If this key isn't specified in the input data, the method falls back to the current date.

### **ccrz.ccApiPriceList.PRGRID**

String that specifies a [price group ID](#) record to filter price lists by. If this key isn't specified in the input data, the method falls back to the value of the `ccrz.cc_CallContext.priceGroupId` field.

**ccrz.ccApiPriceList.STOREFRONT**

String that specifies the name of a storefront for filtering the query.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

SEE ALSO:

[ccrz.cc\\_CallContext](#)

[ccrz\\_\\_E\\_AccountSettings\\_\\_c](#)

[ccrz\\_\\_E\\_PriceGroup\\_\\_c](#)

## ccrz.ccLogicPLEntLists.fetchPriceGroupPriceListIDs

Returns the IDs of price lists associated with the account's price group, if applicable.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchPriceGroupPriceListIDs (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicPLEntLists.createPricingContext` method.

This method requires the following key in the working map:

**ccrz.ccApiPriceList.PRGRID**

String that specifies a [price group ID](#) record to filter price lists by.



**Note:** If the working map doesn't include this key, this method is skipped and execution continues to the `ccrz.ccLogicPLEntLists.fetchAccountGroupPriceListIDs` method.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiPriceList.PLIDS**

Map<String, Set<String>> keyed by `ccrz.ccApiPriceList.PGPLIDS`, which contains the IDs of price lists associated with the account's price group.

SEE ALSO:

[ccrz\\_\\_E\\_PriceGroup\\_\\_c](#)

**ccrz.ccLogicPLEntLists.fetchAccountGroupPriceListIDs**

Returns the IDs of price lists associated with the account's account group, if applicable. This method invokes the `ccrz.ccServiceDAO.doQuery` method for executing the query.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> fetchAccountGroupPriceListIDs (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> from the [ccrz.ccLogicPLEntLists.fetchPriceGroupPriceListIDs](#) method.

This method executes only if either of the following conditions are true:

- The working map doesn't include the `ccrz.ccApiPriceList.PRGRID` key.
- The working map includes `ccrz.ccApiPriceList.ALLWMIX => true`.

**Outputs**

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApiPriceList.PLIDS**

Map<String, Set<String>> keyed by `ccrz.ccApiPriceList.ACCTGRPPLIDS`, which contains the IDs of price lists associated with the account's price group.

SEE ALSO:

[ccrz\\_\\_E\\_AccountGroup\\_\\_c](#)

**ccrz.ccLogicPLEntLists.performRefetch**

Call the `ccrz.ccApiPriceList.fetch` method for fetching the price lists and price list items that correspond to the price list IDs returned from the previous methods.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> performRefetch(Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> from the [ccrz.ccLogicPLEntLists.fetchAccountGroupPriceListIDs](#) method.

This method requires the following key in the working map:

**ccrz.ccApi.SIZING**

Map<String, Object> that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiPriceList.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```



**Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiPriceList.PRICELISTS`

`Map<String, Map<String, Object>>` keyed by the IDs of the entitled price lists.

### `ccrz.ccApiPriceList.PRICELISTITEMS`

`Map<String, Map<String, Object>>` keyed by the IDs of the price list items that belong to the price lists in `ccrz.ccApiPriceList.PRICELISTS`.

#### Note:

- For more information about the `ccrz.ccApiPriceList.PRICELISTS` and `ccrz.ccApiPriceList.PRICELISTITEMS` keys, see the [ccrz.ccApiPriceList.fetch method](#).
- The returned price lists aren't in order. To help you sort the return data, compare the returned price lists with the order of price list IDs returned in parallel in the `ccrz.ccApiPriceList.PLIDS`.

SEE ALSO:

[ccrz.ccApiPriceList.fetch](#)

## ccrz.ccLogicProductFind

Define the business logic for returning and pricing entitled products that match a search string or that belong to a set of specified category IDs.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiProduct.find](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [performProductFind](#)

Calls the `ccrz.ccApiProductIndex.fetchEntitled` method for returning a list of Salesforce IDs of entitled products that match a search string or that belong to a set of specified category IDs.

#### [priceProducts](#)

Calls the `ccrz.ccApiProduct.fetch` method with `ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true` for querying product records and pricing data for the product IDs that match the category or search string.

#### [processProducts](#)

Iterates through the queried products and adds related data, including search rankings, product category assignments, quantity rules, friendly URLs, and more.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicProductFind.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiProduct.find` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicProductFind` class in the following order:

1. `processInput`
2. `performProductFind`
3. `priceProducts`
4. `processProducts`
5. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicProductFind.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicProductFind.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:



**ccrz.ccApiProduct.CATEGORY\_IDS**

Set<String> of specific category IDs whose assigned products you want to return and price.

**ccrz.ccService.SEARCHSTRING**

String to use for searching product records, and whose results you want to return and price.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

The [ccrz.ccLogicProductFind.prepReturn](#) method removes this intermediate key from the working map.

## ccrz.ccLogicProductFind.performProductFind

Calls the `ccrz.ccApiProductIndex.fetchEntitled` method for returning a list of Salesforce IDs of entitled products that match a search string or that belong to a set of specified category IDs.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> performProductFind (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working Map<String, Object> after the [ccrz.ccLogicProductFind.processInput](#) method completes.

This method requires one of the following keys in the working map:

**ccrz.ccApiProduct.CATEGORY\_IDS**

Set<String> of specific category IDs whose assigned products you want to return and price.

**ccrz.ccService.SEARCHSTRING**

String to use for searching product records, and whose results you want to return and price.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### **ccrz.ccApiProduct.CURRENT\_INDEX**

Map<String, Object> that specifies the product index information for the product to use as a starting point.

```
Map<String, Object> findProductsStartingFrom = new Map<String, Object> {
    primary => 'String of product's sfdcName',
    secondary => CC Product Index record ID
}
```

### **ccrz.ccApiProduct.INCLUDE\_COUNT**

Boolean


Value	Usage
<i>true</i>	Return the total count of products, keyed under <code>ccrz.ccApiProduct.COUNT</code> .
<i>false</i> (default)	Don't return product count.

### **ccrz.ccApiProduct.PRODUCT\_FILTERS**

List<Map<String, Object>>, where each Map<String, Object> specifies spec values to filter products by.

### **ccrz.ccApiProduct.PRODUCT\_LIMIT**

Integer that specifies the maximum number of product IDs to return.

 **Note:** If you specify a lower limit than the actual number of results, the return data includes the index information for the next indexed product. For example, if you specify a limit of five, the return data includes the index information for the sixth product that matches the find criteria.

### **ccrz.ccApiProduct.PRODUCTLOCALE**

String that specifies one of your storefront's supported locales. The method returns internationalized product values in the `ccrz.ccApiProduct.PRODUCTLIST` return key. The service layer translates the values from `ccrz__E_ProductItemI18N__c` records and handles fallback languages, if necessary.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

### **ccrz.ccApiProduct.PRODUCTSTOREFRONT**

String that specifies the name of a storefront for filtering the query.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### **ccrz.ccService.ORDERBY**

String that specifies the field to order results by. Possible values include:

- `ccrz.ccApiProductIndex.BY_NAME`
- `ccrz.ccApiProductIndex.BY_PRICE`
- `ccrz.ccApiProductIndex.BY_START_DATE`
- `ccrz.ccApiProductIndex.BY_RELEVANCE`
- `ccrz.ccApiProductIndex.BY_SEQUENCE` (default)

### **ccrz.ccService.SORTDESC**

Boolean

Value	Usage
<code>true</code>	Sort returned products by the field that the <code>ccrz.ccService.ORDERBY</code> key specifies, in descending order.
<code>false</code> (default)	Sort returned products in ascending order.

## Logic Performed

This method passes the input keys to `ccrz.ccApiProductIndex.fetchEntitled`, and parses that method's output.

## Outputs

This method returns the working `Map<String, Object>`, including the `ccrz.ccApiProductIndex.fetchEntitled` outputs.

SEE ALSO:

[ccrz.ccApiProductIndex.fetchEntitled](#)

## ccrz.ccLogicProductFind.priceProducts

Calls the `ccrz.ccApiProduct.fetch` method with `ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true` for querying product records and pricing data for the product IDs that match the category or search string.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> priceProducts (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductFind.performProductFind` method completes.

This method requires the following key in the working map:

**ccrz.ccApiProductIndex.PRODUCT\_IDS**

List<String> of entitled product IDs that the `ccrz.ccApiProductIndex.fetchEntitled` method returned during the `ccrz.ccLogicProductFind.performProductFind` method's execution.

## Logic Performed

This method completes the following actions:

1. Passes `ccrz.ccApiProductIndex.PRODUCT_IDS` for the `ccrz.ccApiProduct.PRODUCTLIST` input for `ccrz.ccApiProduct.fetch`.
2. Specifies the `ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true` key in the input map for `ccrz.ccApiProduct.fetch`.
3. Checks the values of the Show List Price (`p1.dsp1listprc`) and Show Savings (`p1.dsp1svprc`) configurations in the Product List storefront configuration module. When these configurations are TRUE, the method sets `ccrz.ccApiProduct.INCLUDEBASE => true` in the input data.

## Outputs

This method returns the working Map<String, Object>, including the `ccrz.ccApiProduct.fetch` outputs.

SEE ALSO:

[ccrz.ccApiProduct.fetch](#)

[ccrz.ccLogicProductPricing](#)

[Salesforce Help Configure Product Tile Details and Options on the Product List Page](#)

## ccrz.ccLogicProductFind.processProducts

Iterates through the queried products and adds related data, including search rankings, product category assignments, quantity rules, friendly URLs, and more.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processProducts (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductFind.performProductFind` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiProduct.PRODUCTLIST`

List<Object> of `ccrz__E_Product__c` records that the `ccrz.ccApiProduct.fetch` method returned during the `ccrz.ccLogicProductFind.priceProducts` method's execution.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### `ccrz.ccApiProductIndex.PRODUCT_CATEGORY_SEQUENCE`

Map<String, Decimal> that relates each entitled product ID to its corresponding product category sequence value.



**Note:** The `ccrz.ccApiProductIndex.fetchEntitled` method returns this key during the `ccrz.ccLogicProductFind.performProductFind` method's execution.

### `ccrz.ccApiProductIndex.PRODUCT_SEARCH_RANKINGS`

Map<String, Integer> that relates each entitled product ID to its determined relevance rank. For example:

```
{
  "an18digitProductID" : 100,
  "an18digitProductID" : 200,
  "an18digitProductID" : 250,
  "an18digitProductID" : 50
}
```



**Note:** The `ccrz.ccApiProductIndex.fetchEntitled` method returns this key during the `ccrz.ccLogicProductFind.performProductFind` method's execution.

## Logic Performed

For each product in `ccrz.ccApiProduct.PRODUCTLIST`, this method completes the following actions:

1. Checks for the product's ID in `ccrz.ccApiProductIndex.PRODUCT_SEARCH_RANKINGS`, and adds the search ranking to the product. Otherwise, this method finds the product's ID in `ccrz.ccApiProductIndex.PRODUCT_CATEGORY_SEQUENCE`, and adds the category sequence to the product.
2. Evaluates the product's `ccrz__ProductStatus__c` field. If the status is Released, this method sets a Boolean, `canAddToCart`, that indicates that the product can be added to the cart.



**Note:** For more information about product status, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_product\\_create.htm](https://help.salesforce.com/articleView?id=b2b_commerce_product_create.htm).

3. Invokes the `ccrz.cc_api_ProductQuantityRule` extension point class for evaluating whether any applicable quantity rules exist for the product.
4. Calls the `ccrz.ccApiProduct.getUrlFor` method for returning the product's friendly URL.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiProduct.PRODUCTLIST**

List<Object> of `ccrz__E_Product__c` records passed into this method and updated with the results of this method's evaluations.

## SEE ALSO:

[ccrz.ccApiProduct.fetch](#)

[ccrz.ccApiProduct.getUrlFor](#)

[ccrz.ccApiProductIndex.fetchEntitled](#)

[ccrz.cc\\_api\\_ProductQuantityRule](#)

**ccrz.ccLogicProductFind.prepReturn**

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductFind.processProducts` method.

**Outputs**

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicProductFindFilters

Define the business logic for returning filters and spec values for entitled products that match a search string or that belong to a set of specified category IDs.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Global API Caller

[ccrz.ccApiProduct.findFilters](#)

### Methods

#### IN THIS SECTION:

##### [process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

##### [processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

##### [fetchEntitledProducts](#)

Calls the `ccrz.ccApiProductIndex.fetchEntitled` method for returning a list of Salesforce IDs of entitled products that match a search string or that belong to a set of specified category IDs.

##### [fetchFilters](#)

Calls the `ccrz.ccApiSpec.fetchProductSpecIndices` method for returning `ccrz__E_ProductSpecIndex__c` records that correspond to the entitled products.

##### [processFilterData](#)

Aggregates `ccrz__E_ProductSpecIndex__c` filter data, keyed by spec.

##### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicProductFindFilters.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiProduct.findFilters` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicProductFindFilters` class in the following order:

1. `processInput`
2. `fetchEntitledProducts`
3. `fetchFilters`
4. `processFilterData`
5. `prepReturn`

### Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicProductFindFilters.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicProductFindFilters.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

### **ccrz.ccApiProduct.CATEGORY\_IDS**

`Set<String>` of specific category IDs whose assigned products you want to return product filters for.

### **ccrz.ccService.SEARCHSTRING**

String to use for searching product index records for returning product filters.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicProductFindFilters.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicProductFindFilters.fetchEntitledProducts

Calls the `ccrz.ccApiProductIndex.fetchEntitled` method for returning a list of Salesforce IDs of entitled products that match a search string or that belong to a set of specified category IDs.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchEntitledProducts (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductFindFilters.processInput` method completes.

This method requires one of the following keys in the working map:

### **ccrz.ccApiProduct.CATEGORY\_IDS**

`Set<String>` of specific category IDs whose assigned products you want to return product filters for.

### **ccrz.ccService.SEARCHSTRING**

String to use for searching product index records for returning product filters.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### **ccrz.ccApiProduct.PRODUCT\_FILTERS**

`List<Map<String, Object>>`, where each `Map<String, Object>` specifies a spec ID and product spec value to filter products by:

```
[ {
  "specValues" : [ {
    "value" : "Blue"
  } ],
  "sfid" : "alh6A00000045s5QAA"
} ]
```

### **ccrz.ccApiProduct.PRODUCTLOCALE**

String that specifies one of your [storefront's supported locales](#).



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

### **ccrz.ccApiProduct.PRODUCTSTOREFRONT**

String that specifies the name of a storefront for filtering the query.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Logic Performed

This method passes the input keys to `ccrz.ccApiProductIndex.fetchEntitled`, and parses that method's output.

## Outputs

This method returns the working `Map<String, Object>`, including the following `ccrz.ccApiProductIndex.fetchEntitled` output:

### `ccrz.ccApiProductIndex.PRODUCT_IDS`

`List<String>` of entitled product IDs that the `ccrz.ccApiProductIndex.fetchEntitled` method returned.

SEE ALSO:

[ccrz.ccApiProductIndex.fetchEntitled](#)

## `ccrz.ccLogicProductFindFilters.fetchFilters`

Calls the `ccrz.ccApiSpec.fetchProductSpecIndices` method for returning `ccrz__E_ProductSpecIndex__c` records that correspond to the entitled products.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchFilters (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductFindFilters.fetchEntitledProducts` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiProductIndex.PRODUCT_IDS`

`List<String>` of entitled product IDs that the `ccrz.ccApiProductIndex.fetchEntitled` method returned during the `ccrz.ccLogicProductFindFilters.fetchEntitledProducts` method's execution.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### **ccrz.ccApiProduct.PRODUCTLOCALE**

String that specifies one of your storefront's supported locales.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

## Logic Performed

This method passes the input keys to `ccrz.ccApiSpec.fetchProductSpecIndices`, and parses that method's output.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiProduct.PRODUCT\_FILTERS**

`List<Map<String, Object>>` that corresponds to the `ccrz.ccApiSpec.PRODUCT_SPEC_INDICES` output from `ccrz.ccApiSpec.fetchProductSpecIndices`, where each `Map<String, Object>` specifies a spec ID and product spec value:

```
[
  {
    "specValues": [
      {
        "count": 100,
        "value": "SpecValA"
      }
    ],
    "filterType": "Checkbox",
    "sequence": 200,
    "sfid": "alh6A000001ERMvQAO",
    "name": "SpecA0"
  }
]
```

SEE ALSO:

[ccrz\\_\\_E\\_ProductSpecIndex\\_\\_c](#)

[ccrz.ccApiSpec.fetchProductSpecIndices](#)

## ccrz.ccLogicProductFindFilters.processFilterData

Aggregates `ccrz__E_ProductSpecIndex__c` filter data, keyed by spec.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processFilterData(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductFindFilters.fetchFilters` method completes.

This method requires the following key in the working map:

### `ccrz.ccApiProduct.PRODUCT_FILTERS`

`List<Map<String, Object>>`, where each `Map<String, Object>` specifies a spec ID and product spec value:

```
[
  {
    "specValues": [
      {
        "count": 100,
        "value": "SpecValA"
      }
    ],
    "filterType": "Checkbox",
    "sequence": 200,
    "sfid": "alh6A000001ERMvQAO",
    "name": "SpecA0"
  }
]
```

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiProduct.PRODUCT_FILTERS`

`Map<String, Map<String, Object>>` of aggregated product filters, keyed by spec value. When serialized to JSON, this map of filters uses the following format:

```
[
  {
    "specValues": [
      {
        "count": 100,
        "value": "SpecValA"
      }
    ]
  }
]
```

```

    ],
    "filterType": "Checkbox",
    "sequence": 200,
    "sfid": "alh6A000001ERMvQAO",
    "name": "SpecA0"
  }
]

```

## ccrz.ccLogicProductFindFilters.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductFindFilters.processFilterData` method.

### Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicProductPricing

Define the business logic for pricing products.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiProduct.fetch` when the input data specifies `ccrz.ccApiProduct.PARAM_INCLUDE_PRICING => true`.

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

#### [processInputData](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

#### [processProductList](#)

Determines the complete list of entitled products for the pricing evaluation, including component products that belong to composite products such as bundles and dynamic kits.

#### [fetchPriceListsAndItems](#)

Calls the `ccrz.ccApiPriceList.fetchEntitled` and `ccrz.ccApiPriceList.fetchPriceListItems` methods for returning the entitled price list items for the set of product IDs. This method doesn't query price list items for the storefront's base price list, or query tiered pricing or attribute pricing JSON data. Later methods in the logic chain execute those queries.

#### [processPriceListItems](#)

Organizes price list items data by sellers, subscriptions, and entitlement.

#### [fetchBasePriceListsAndItems](#)

Calls the `ccrz.ccApiPriceList.fetch` method for returning the storefront's base price list and price list items. This query is separate from the `ccrz.ccLogicProductPricing.fetchPriceListsAndItems` method because a buyer's entitled price list can be the same price list as the base price list. This logic chain maintains separate data sets for entitled and base price list data.

#### [evaluatePricing](#)

Loops through each product and calls an appropriate inner method for pricing the product based on its product type.

#### [evaluateBasicPricing](#)

Evaluates criteria for pricing a standard product, including entitled price lists, base price lists, seller data, and subscription data. This method also calculates a buyer's savings from an entitled price list compared to a base price list.

### [evaluateBundlePricing](#)

Evaluates criteria for pricing a bundle product, including entitled price lists, base price lists, and subscription data. A bundle product doesn't support sellers. This method also calculates a buyer's savings from an entitled price list compared to a base price list.

### [evaluateKitPricing](#)

Evaluates criteria for pricing a kit product, including entitled price lists, base price lists, and subscription data. A kit product doesn't support sellers.

### [evaluateDynamicKitPricing](#)

Evaluates criteria for pricing a dynamic kit product, including entitled price lists, base price lists, and subscription data. A dynamic kit product doesn't support sellers.

### [evaluateAggregatePricing](#)

Evaluates criteria for pricing an aggregated product, including entitled price lists and base price lists. An aggregated product doesn't support sellers or subscriptions.

### [evaluateAssemblyPricing](#)

Evaluates criteria for pricing an assembly product, including entitled price lists and base price lists. An assembly product doesn't support sellers or subscriptions.

### [evaluateCustomProductPricing](#)

Provides an extension point for evaluating criteria for pricing a product type other than the default types.

### [evaluateSellersPricing](#)

Evaluates criteria for pricing a standard product associated with a seller.

### [evaluateSubProdTermPricing](#)

Evaluates criteria for pricing a standard product, bundle, or dynamic kit associated with a subscription.

### [evaluateKitProductSubscriptions](#)

Evaluates criteria for pricing a kit associated with a subscription.

### [determinePrice](#)

Sets a price for any product when called from another evaluation method. This method is the primary logic for applying a price from an entitled price list, base price list, seller, or subscription. This method also evaluates any price list items associated with a contract pricing's price group, which take priority over an entitled price list.

### [processSelectedPLIs](#)

Queries attributes or tiers pricing data for the price list items whose prices the `ccrz.ccLogicProductPricing.determinePrice` method applied.

### [processPricingTiersForPLI](#)

Evaluates product pricing data for an individual price list item and calculates the unit price.

### [processAttributes](#)

Queries attributes data processed by the `ccrz.ccLogicProductPricing.processSelectedPLIs` method.

### [prepReturnData](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## **ccrz.ccLogicProductPricing.process**

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiProduct.fetch` method.

## Logic Performed

This method invokes other methods of the `ccrz.ccLogicProductPricing` class in the following order:

1. `processInputData`
2. `processProductList`
3. `fetchPriceListsAndItems`
4. `processPriceListItems`
5. `fetchBasePriceListsAndItems`
6. `evaluatePricing`, which calls the appropriate inner method for pricing the product based on its product type:
  - `evaluateBasicPricing`, which calls into one of the following inner methods:
    - `evaluateSellersPricing`, which calls into `determinePrice`
    - `evaluateSubProdTermPricing`, which calls into `determinePrice`
    - `determinePrice`, which can call into `processPricingTiersPLI`
  - `evaluateBundlePricing`, which calls into one of the following inner methods:
    - `determinePrice`
    - `evaluateSubProdTermPricing`, which calls into `determinePrice`
  - `evaluateKitPricing`, which calls into one of the following inner methods:
    - `determinePrice`
    - `evaluateKitProductSubscriptions`, which calls into `determinePrice`
  - `evaluateDynamicKitPricing`, which calls into one of the following inner methods:

- determinePrice
  - evaluateSubProdTermPricing, which calls into determinePrice
  - evaluateAggregatePricing, which calls into determinePrice
  - evaluateAssemblyPricing, which calls into determinePrice
  - evaluateCustomProductPricing
7. processSelectedPLIs, which can call into processPricingTiersPLI
  8. processAttributes
  9. prepReturnData

## Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.



**Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicProductPricing.processInputData

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInputData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicProductPricing.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

**ccrz.ccApiProduct.PRODUCTLIST**

List<Map<String, Object>>, where each Map<String, Object> represents a `ccrz__E_Product__c` that the `ccrz.ccServiceProduct` data service provider returned for the `ccrz.ccApiProduct.fetch` call.

**Logic Performed**

This method evaluates the input data and sets required key values. This method applies fallback values for any values that the input data doesn't specify.

**Outputs**

This method returns a Map<String, Object>, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

**ccrz.ccApiProduct.COMPPARENTTYPES**

Set<String> that's initially empty. Later methods in the logic chain evaluate this key.

**ccrz.ccApiProduct.CURRCODE**

String that specifies an ISO 4217 currency code, such as *USD* or *JPY*, for filtering price list items. By default, this method sets the value from `ccrz.cc_CallContext.userCurrency` unless the input data explicitly specifies a different value.

**ccrz.ccApiProduct.EFFDATE**

Date for filtering by price list item Start Date and End Date values. If the input data doesn't specify a date, this method falls back to `System.now()`.

**ccrz.ccApiProduct.EVALUATIONMODE**

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.




**Note:** If the input data doesn't specify this value, this method falls back to `ccrz.cc_CallContext.entAccountGroup.PriceListSelectionMethod__c`.

**ccrz.ccApiProduct.INCLUDE\_ATTRIBUTE\_PRICING**

Boolean


Value	Usage
true	Query for <code>ccrz__E_Attribute__c</code> records that are related to each product, and return the attribute pricing JSON applied to the product's entitled price list item.
false (default)	Don't return any attribute pricing.

 **Note:** When this value is `false` but `ccrz.ccApiProduct.INCLUDE_TIER_PRICING` is `true`, the logic chain queries tiered pricing JSON data.

#### `ccrz.ccApiProduct.INCLUDE_TIER_PRICING`

Boolean

Value	Usage
<code>true</code>	Query tiered pricing JSON. For more information about tiered pricing, see <a href="https://help.salesforce.com/articleView?id=sf.b2b_commerce_tiered_pricing.htm">https://help.salesforce.com/articleView?id=sf.b2b_commerce_tiered_pricing.htm</a> .
<code>false</code> (default)	Don't return any tiered pricing.

 **Note:** When this value is `false` but `ccrz.ccApiProduct.INCLUDE_ATTRIBUTE_PRICING` is `true`, the logic chain queries attribute pricing JSON data.

#### `ccrz.ccApiProduct.INCLUDEBASE`

Boolean

Value	Usage
<code>true</code> (default)	Evaluate the base price list. By default, this method returns <code>true</code> unless the input data explicitly sets <code>false</code> for this key.
<code>false</code>	Ignore the base price list.

#### `ccrz.ccApiProduct.PARAM_INCLUDE_SELLERS`

Boolean

Value	Usage
<code>true</code> (default)	Evaluate seller price lists. By default, this method returns <code>true</code> unless the input data explicitly sets <code>false</code> for this key.
<code>false</code>	Skip seller price lists.

#### `ccrz.ccApiProduct.PARSE_ATTRIBUTE_PRICING`

Boolean

Value	Usage
<code>true</code> (default)	Parse the attribute pricing JSON. By default, this method returns <code>true</code> unless the input data explicitly sets <code>false</code> for this key.
<code>false</code>	Skip attribute pricing.

#### `ccrz.ccApiProduct.PRODUCTIDQTYMAP`

`Map<String, Object>` that maps product IDs to their quantities to price.

**ccrz.ccApiProduct.SELLERS**

Set<String> of Salesforce IDs `ccrz__E_Seller__c` records for filtering price list items.

**ccrz.ccApiSubProdTerm.SPTBYPRODID**

Map<String, Object> that maps subscription terms to their product IDs.

**ccrz.ccApiSubProdTerm.SUBPRODTERMS**

Set<String> of Salesforce IDs for subscription terms

**ccrz.ccLogic.ORIGINAL\_INPUT**

Map<String, Object> of the original, unmodified input data.

**'prodPLIsToFullyQuery'**

Set<String> that's initially empty. Later methods in the logic chain evaluate this key.

**'selectedPLIs'**

Set<String> that's initially empty. Later methods in the logic chain evaluate this key.

## ccrz.ccLogicProductPricing.processProductList

Determines the complete list of entitled products for the pricing evaluation, including component products that belong to composite products such as bundles and dynamic kits.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processProductList (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductPricing.processInputData` method completes.

This method requires the following keys in the working map:

**ccrz.ccApiProduct.PRODUCTLIST**

List<`ccrz__E_Product__c`> that the `ccrz.ccServiceProduct` data service provider returned for the original `ccrz.ccApiProduct.fetch` call.

**'prodPLIsToFullyQuery'**

Set<String> that's initially empty.

## Logic Performed

This method completes the following actions:

1. Evaluates `ccrz.ccApiProduct.PRODUCTLIST` and add each product ID to a set of IDs, which later methods use for querying price list items.
2. For each composite product included in the `compositeProductsS` field of `ccrz.ccApiProduct.PRODUCTLIST`, this method adds the ID of each component product to the set of IDs.
3. When a composite product is a kit, this method adds its composite product ID to `'prodPLIsToFullyQuery'`, which later methods use for querying price list items.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiPriceList.PRODUCTIDS**

Set<String> of Salesforce IDs for all products to price, including component products that belong to composite products.

**'prodPLIsToFullyQuery'**

Set<String> of Salesforce IDs of kit products to price. These IDs are in a separate set for querying tiered pricing information for the kit.

SEE ALSO:

[ccrz.ccApiProduct.fetch](#)

## ccrz.ccLogicProductPricing.fetchPriceListsAndItems

Calls the `ccrz.ccApiPriceList.fetchEntitled` and `ccrz.ccApiPriceList.fetchPriceListItems` methods for returning the entitled price list items for the set of product IDs. This method doesn't query price list items for the storefront's base price list, or query tiered pricing or attribute pricing JSON data. Later methods in the logic chain execute those queries.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchPriceListsAndItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductPricing.processProductList` method completes.

This method requires the following keys in the working map:

### **ccrz.ccApi.API\_VERSION**

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### **ccrz.ccApiPriceList.PRODUCTIDS**

`Set<String>` of Salesforce IDs for all products to price, including component products that belong to composite products.

### **ccrz.ccApiProduct.CURRCODE**

String that specifies an ISO 4217 currency code, such as *USD* or *JPY*, for filtering price list items.

### **ccrz.ccApiProduct.EFFDATE**

Date for filtering by price list item Start Date and End Date values.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

### **ccrz.ccApiProduct.SELLERS**

`Set<String>` of Salesforce IDs `ccrz__E_Seller__c` records for filtering price list items.

### **ccrz.ccApiSubProdTerm.SUBPRODTERMS**

`Set<String>` of Salesforce IDs for subscription terms

### **'prodPLIsToFullyQuery'**

`Set<String>` of Salesforce IDs of kit products to price. These IDs are in a separate set for querying tiered pricing information for the kit.

## Logic Performed

This method completes the following actions:

1. Passes `ccrz.ccApiPriceList.PRODUCTIDS` and other available input keys (except `'prodPLIsToFullyQuery'`) to `ccrz.ccApiPriceList.fetchEntitled`. By default, this query applies the following [data size request parameters](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAttribute.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XL
    },
    ccrz.ccApiPriceList.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    },
    ccrz.ccApiPriceList.ENTITYNAME_PLI => new Map<String, Object> {
```

```

        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XL
    }
}

```

2. Parses the `ccrz.ccApiPriceList.fetchEntitled` return data for the returned price list items.
3. Passes '`prodPLIsToFullyQuery`' and other available input keys (except `ccrz.ccApiPriceList.PRODUCTIDS`) to `ccrz.ccApiPriceList.fetchPriceListItems`. By default, this query applies the following [data size request parameters](#):

```

ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiPriceList.ENTITYNAME_PLI => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XL
    }
}

```

4. Parses the `ccrz.ccApiPriceList.fetchPriceListItems` return data for the returned price list items, and adds them to the list of price list items returned from `ccrz.ccApiPrice.fetchEntitled`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiPriceList.PRICELISTITEMS**

`Map<String, Object>` of `ccrz__E_PriceListItem__c` data, keyed by ID, that the `ccrz.ccApiPrice.fetchEntitled` and `ccrz.ccApiPriceList.fetchPriceListItems` methods returned.

### **ccrz.ccApiPriceList.PRICELISTS**

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID, that the `ccrz.ccApiPriceList.fetchEntitled` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_Pricelist\\_\\_c](#)

[ccrz\\_\\_E\\_PricelistItem\\_\\_c](#)

[ccrz.ccApiPriceList.fetchEntitled](#)

[ccrz.ccApiPriceList.fetchPriceListItems](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccLogicProductPricing.processPricelistItems

Organizes price list items data by sellers, subscriptions, and entitlement.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processPriceListItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductPricing.fetchPriceListsAndItems` method completes.

This method requires the following keys in the working map:

### `ccrz.ccApiPriceList.PRICELISTITEMS`

`Map<String, Object>` of `ccrz__E_PriceListItem__c` data, keyed by ID.

### `ccrz.ccApiPriceList.PRICELISTS`

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiProduct.PARAM_INCLUDE_SELLERS`

Boolean

Value	Usage
<code>true</code> (default)	Evaluate seller price lists.
<code>false</code>	Skip seller price lists.

## Logic Performed

This method completes the following actions:

1. Instantiates an empty `Map<String, Map<String, List<Map<String, Object>>>>` for organizing price list items
2. Evaluates each `ccrz__E_PriceListItem__c` in `ccrz.ccApiPriceList.PRICELISTITEMS`, and completes the following actions:
  - a. If the price list item is associated with a seller, this method adds the item to a `List<Map<String, Object>>`, which groups price list items associated with sellers. This method adds the list to a `Map<String, List>`, keyed by a string that indicates that the list includes seller-related price list items.
  - b. If the item is associated with a subscription, this method adds the item to the `List<Map<String, Object>>`, which groups price list items associated with subscriptions. This method adds the list to a `Map<String, List>`, keyed by a string that indicates that the list includes subscription-related price list items.

- c. Otherwise, this method adds the item to the `List<Map<String, Object>>`, which groups entitled price list items. This method adds the list to a `Map<String, List>`, keyed by a string that indicates that the list includes entitled price list items.

3. Adds each second-level `Map`, keyed by price list item type, to the outer `Map`, keyed by product ID.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiPriceList.PRICELISTITEMS`

`Map<String, Map<String, List<Map<String, Object>>>>` where:

- The outer `Map` is keyed by a string that specifies a product ID.
- Each second-level `Map` is keyed by a string that specifies a type of price list item: `'entitledPLIs'`, `'sellersPLIs'`, or `'subProdTermPLIs'`.
- Each inner `List<Map<String, Object>>` is a list of entitled, seller, or subscription price list items.



**Note:** This key replaces the `ccrz.ccApiPriceList.PRICELISTITEMS` input.

SEE ALSO:

[ccrz\\_\\_E\\_Pricelist\\_\\_c](#)

[ccrz\\_\\_E\\_PricelistItem\\_\\_c](#)

[Salesforce Help Manage Prices for Storefront Products with Price Lists](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## `ccrz.ccLogicProductPricing.fetchBasePriceListsAndItems`

Calls the `ccrz.ccApiPriceList.fetch` method for returning the storefront's base price list and price list items. This query is separate from the `ccrz.ccLogicProductPricing.fetchPriceListsAndItems` method because a buyer's entitled price list can be the same price list as the base price list. This logic chain maintains separate data sets for entitled and base price list data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchBasePriceListsAndItems (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicProductPricing.processPriceListItems` method completes.

This method requires the following keys in the working map:

### `ccrz.ccApi.API_VERSION`

The [version of the B2B Commerce for Visualforce API](#) to reference for the method call. We recommend that you use the `ccrz.ccApi.CURRENT_VERSION` constant whenever possible, and only reference a specific version for compatibility if necessary.



**Note:** If this key isn't specified, the method returns a `ccrz.ccApi.NoApiVersionException`.

### `ccrz.ccApiPriceList.INCLSELLERS`

Boolean set to `false`.

### `ccrz.ccApiPriceList.NAME`

String that specifies the name of the [storefront's base price list](#). This method references the value from `ccrz.cc_CallContext.basePriceList`.

### `ccrz.ccApiPriceList.PRODUCTIDS`

Set<String> of Salesforce IDs for all products to price, including component products that belong to composite products.

### `ccrz.ccApiProduct.CURRCODE`

String that specifies an ISO 4217 currency code, such as `USD` or `JPY`, for filtering price list items.

### `ccrz.ccApiProduct.EFFDATE`

Date for filtering by price list item Start Date and End Date values.

## Logic Performed

This method completes the following actions:

1. Passes the input keys to `ccrz.ccApiPriceList.fetch`. By default, this query applies the following [data size request parameters](#):

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiAttribute.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_XL
    },
    ccrz.ccApiPriceList.ENTITYNAME_PLI => new Map<String, Object> {
        ccrz.ccApi.SZ_DATA => ccrz.ccApi.SZ_L
    }
}
```

2. Parses the `ccrz.ccApiPriceList.PRICELISTITEMS` output from `ccrz.ccApiPriceList.fetch`, and builds a map of price list items in the base price list, keyed by product ID.



**Note:** This map excludes subscription-related price list items in the base price list.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### 'basePriceListItems'

`Map<String, Object>` of `ccrz__E_PriceListItem__c` data, keyed by ID, that the `ccrz.ccApiPrice.fetch` method returned.

### 'basePriceLists'

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID, that the `ccrz.ccApiPrice.fetch` method returned.

SEE ALSO:

[ccrz\\_\\_E\\_PricelistItem\\_\\_c](#)

[ccrz.ccApiPriceList.fetch](#)

[Salesforce Help Specify Your Storefront's Base Price List](#)

## ccrz.ccLogicProductPricing.evaluatePricing

Loops through each product and calls an appropriate inner method for pricing the product based on its product type.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> evaluatePricing (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the [ccrz.ccLogicProductPricing.fetchBasePriceListsAndItems](#) method completes.

This method requires the following key in the working map:

### **ccrz.ccApiProduct.PRODUCTLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_Product__c` that the `ccrz.ccServiceProduct` data service provider returned for the original `ccrz.ccApiProduct.fetch` call.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### **ccrz.ccApiSubProdTerm.SPTBYPRODID**


Map<String, Object> that maps subscription terms to their product IDs.

## Logic Performed

This method completes the following actions for each product in `ccrz.ccApiProduct.PRODUCTLIST`:

1. Instantiates an intermediate key, 'product', for the `ccrz__E_Product__c` record to evaluate.
2. Evaluates whether subscription terms exist for the product, and adds subscription terms data to an intermediate Map<String, Object>, 'subProdTerms'.
3. Calls the appropriate inner method for pricing the product, depending on its product type. If subscription terms exist for the product, this method passes 'subProdTerms' to the inner method.

Product Type	Method
Standard product	<code>evaluateBasicPricing</code>
Bundle	<code>evaluateBundlePricing</code>
Kit	<code>evaluateKitPricing</code>
Dynamic Kit	<code>evaluateDynamicKitPricing</code>
Aggregated	<code>evaluateAggregatePricing</code>
Assembly	<code>evaluateAssemblyPricing</code>
Other	<code>evaluateCustomProductPricing</code>

 **Note:** If the appropriate inner method doesn't return any price data for the product, this method determines that the buyer isn't entitled to the product. The method removes the product from `ccrz.ccApiProduct.PRODUCTLIST`, which excludes the product from further evaluation in this logic chain.

## Outputs

This method returns the working Map<String, Object> after the inner loop methods complete, with the following key set:

### **ccrz.ccApiProduct.PRODUCTLIST**

List<Map<String, Object>> that corresponds to the `ccrz.ccApiProduct.PRODUCTLIST` input, updated to include only the buyer's entitled `ccrz__E_Product__c` data.

### **ccrz.ccApiProduct.PRODUCTPRICINGDATA**

Map<String, Object> of product pricing data.

This method also removes the following intermediate keys from the working map:

- `ccrz.ccApiPriceList.PRICELISTITEMS`
- `ccrz.ccApiPriceList.PRICELISTS`
- 'product'

- 'subProdTerms'

## ccrz.ccLogicProductPricing.evaluateBasicPricing

Evaluates criteria for pricing a standard product, including entitled price lists, base price lists, seller data, and subscription data. This method also calculates a buyer's savings from an entitled price list compared to a base price list.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> evaluateBasicPricing(Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicProductPricing.evaluatePricing` method.

This method requires the following keys in the working map:

#### **ccrz.ccApiPriceList.PRICELISTITEMS**

`Map<String, Map<String, List<Map<String, Object>>>>` where:

- The outer `Map` is keyed by a string that specifies a product ID.
- Each second-level `Map` is keyed by a string that specifies a type of price list item: 'entitledPLIs', 'sellersPLIs', or 'subProdTermPLIs'.
- Each inner `List<Map<String, Object>>` is a list of entitled, seller, or subscription price list items.

#### **ccrz.ccApiPriceList.PRICELISTS**

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

#### **ccrz.ccApiProduct.EVALUATIONMODE**

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.

Value	Usage
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.

**'basePriceListItems'**

`Map<String, Object>` of `ccrz__E_PriceListItem__c` data, keyed by ID, that belong to the storefront's base price list.

**'completeInputData'**

`Map<String, Object>` that contains a copy of all the input keys and values passed to this method.

**'product'**

`ccrz__E_Product__c` to price.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**`ccrz.ccApiAttribute.ATTRIBUTEIDS`**

`Map<String, Object>` where each key is a parent attribute ID and each value is a `List<String>` of child attribute IDs:

```
new Map<String, Object>{
    'ParentAttributeID1' => new List<String>{'ChildAttributeID1', 'ChildAttributeID2',
    'ChildAttribute3'},
    'ParentAttributeID2' => new List<String>{'ChildAttributeID4', 'ChildAttributeID6',
    'ChildAttribute5'}
}
```

**`ccrz.ccApiProduct.INCLUDE_ATTRIBUTE_PRICING`**

Boolean

Value	Usage
<code>true</code>	Query for <code>ccrz__E_Attribute__c</code> records that are related to each product, and return the attribute pricing JSON applied to the product's entitled price list item.
<code>false</code> (default)	Don't return any attribute pricing.

**`ccrz.ccApiProduct.PARSE_ATTRIBUTE_PRICING`**

Boolean

Value	Usage
<code>true</code> (default)	Parse the attribute pricing JSON. By default, this method returns <code>true</code> unless the input data explicitly sets <code>false</code> for this key.
<code>false</code>	Skip attribute pricing.

**ccrz.ccApiSubProdTerm.SPTBYPRODID**

Map<String, Object> that maps subscription terms to their product IDs.

## Logic Performed

This method completes the following actions:

1. If any sellers data exists for the product, this method calls the `ccrz.ccLogicProductPricing.evaluateSellersPricing` method to handle the sellers pricing.
2. If any subscription data exists for the product, this method calls the `ccrz.ccLogicProductPricing.evaluateSubProdTermPricing` inner method to handle the subscription pricing.
3. To evaluate product's pricing in the buyer's entitled price list, this method calls the `ccrz.ccLogicProductPricing.determinePrice` inner method.
4. To evaluate product's pricing in the storefront's base price list, this method calls the `ccrz.ccLogicProductPricing.determinePrice` inner method.
5. Calculates the buyer's savings from the entitled price list compared to the base price list.

## Outputs

This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**

Map<String, Object> of product pricing data, keyed by product ID, that results from the inner method calls.

SEE ALSO:

[Salesforce Help Create a Product for Your Storefront](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccLogicProductPricing.evaluateBundlePricing

Evaluates criteria for pricing a bundle product, including entitled price lists, base price lists, and subscription data. A bundle product doesn't support sellers. This method also calculates a buyer's savings from an entitled price list compared to a base price list.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



## Signature

```
global virtual Map<String, Object> evaluateBundlePricing(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicProductPricing.evaluatePricing` method.

This method requires the following keys in the working map:

### `ccrz.ccApiPriceList.PRICELISTITEMS`

`Map<String, Map<String, List<Map<String, Object>>>>` where:

- The outer `Map` is keyed by a string that specifies a product ID.
- Each second-level `Map` is keyed by a string that specifies a type of price list item: `'entitledPLIs'` or `'subProdTermPLIs'`.
- Each inner `List<Map<String, Object>>` is a list of entitled or subscription price list items.

### `ccrz.ccApiPriceList.PRICELISTS`

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

### `ccrz.ccApiProduct.EVALUATIONMODE`

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.

### 'basePriceListItems'

`Map<String, Object>` of `ccrz__E_PriceListItem__c` data, keyed by ID, that belong to the storefront's base price list.

### 'completeInputData'

`Map<String, Object>` that contains a copy of all the input keys and values passed to this method.

### 'product'

`ccrz__E_Product__c` to price.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiSubProdTerm.SPTBYPRODID`

`Map<String, Object>` that maps subscription terms to their product IDs.

## Logic Performed

This method completes the following actions:

1. For each component product included in a bundle's `compositeProductsS` related list, this method completes the following actions:
  - a. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from an entitled price list.
  - b. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from the storefront's base price list.
2. For the bundle product itself, this method completes the following actions:
  - a. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the bundle's price from an entitled price list.
  - b. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the bundle's price from the storefront's base price list.
  - c. If any subscription data exists for the product, this method calls the `ccrz.ccLogicProductPricing.evaluateSubProdTermPricing` inner method for handling the bundle's subscription pricing.
3. Calculates the buyer's savings from the entitled price list compared to the base price list.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiProduct.PRODUCTPRICINGDATA`

`Map<String, Object>` of product pricing data, keyed by product ID, that results from the inner method calls.

SEE ALSO:

[Salesforce Help Create a Bundle](#)

## `ccrz.ccLogicProductPricing.evaluateKitPricing`

Evaluates criteria for pricing a kit product, including entitled price lists, base price lists, and subscription data. A kit product doesn't support sellers.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> evaluateKitPricing(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicProductPricing.evaluatePricing` method.

This method requires the following keys in the working map:

### `ccrz.ccApiPriceList.PRICELISTITEMS`

`Map<String, Map<String, List<Map<String, Object>>>>` where:

- The outer `Map` is keyed by a string that specifies a product ID.
- Each second-level `Map` is keyed by a string that specifies a type of price list item: `'entitledPLIs'` or `'subProdTermPLIs'`.
- Each inner `List<Map<String, Object>>` is a list of entitled or subscription price list items.

### `ccrz.ccApiPriceList.PRICELISTS`

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

### `ccrz.ccApiProduct.EVALUATIONMODE`

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.

### 'basePriceListItems'

`Map<String, Object>` of `ccrz__E_PriceListItem__c` data, keyed by ID, that belong to the storefront's base price list.

### 'completeInputData'

`Map<String, Object>` that contains a copy of all the input keys and values passed to this method.

### 'product'

`ccrz__E_Product__c` to price.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiSubProdTerm.SPTBYPRODID`

`Map<String, Object>` that maps subscription terms to their product IDs.

## Logic Performed

This method completes the following actions:

1. For each component product included in a kit's `compositeProductsS` related list, this method completes the following actions:
  - a. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from an entitled price list.
  - b. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from the storefront's base price list.
2. For the kit product itself, this method completes the following actions:
  - a. Calculates the subtotal of all component products' prices from an entitled price list.
  - b. Calculates the subtotal of all component products' prices from the storefront's base price list.
  - c. If any subscription data exists for the kit, this method calls the `ccrz.ccLogicProductPricing.evaluateKitProductSubscriptions` inner method for handling the kit's subscription pricing.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiProduct.PRODUCTPRICINGDATA`

`Map<String, Object>` of product pricing data, keyed by product ID, that results from the inner method calls.

SEE ALSO:

[Salesforce Help Create a Kit](#)

## `ccrz.ccLogicProductPricing.evaluateDynamicKitPricing`

Evaluates criteria for pricing a dynamic kit product, including entitled price lists, base price lists, and subscription data. A dynamic kit product doesn't support sellers.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> evaluateDynamicKitPricing (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicProductPricing.evaluatePricing` method.

This method requires the following keys in the working map:

### `ccrz.ccApiPriceList.PRICELISTITEMS`

`Map<String, Map<String, List<Map<String, Object>>>>` where:

- The outer `Map` is keyed by a string that specifies a product ID.
- Each second-level `Map` is keyed by a string that specifies a type of price list item: `'entitledPLIs'` or `'subProdTermPLIs'`.
- Each inner `List<Map<String, Object>>` is a list of entitled or subscription price list items.

### `ccrz.ccApiPriceList.PRICELISTS`

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

### `ccrz.ccApiProduct.EVALUATIONMODE`

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.

### `'basePriceListItems'`

`Map<String, Object>` of `ccrz__E_PriceListItem__c` data, keyed by ID, that belong to the storefront's base price list.

### `'completeInputData'`

`Map<String, Object>` that contains a copy of all the input keys and values passed to this method.

### `'product'`

`ccrz__E_Product__c` to price.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

### `ccrz.ccApiSubProdTerm.SPTBYPRODID`

`Map<String, Object>` that maps subscription terms to their product IDs.

## Logic Performed

This method completes the following actions:

1. For each component product included in a dynamic kit's `compositeProductsS` related list, this method completes the following actions:
  - a. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from an entitled price list.

- b. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from the storefront's base price list.
2. For the dynamic kit product itself, this method completes the following actions:
  - a. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the dynamic kit's price from an entitled price list.
  - b. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the dynamic kit's price from the storefront's base price list.
  - c. If any subscription data exists for the kit, this method calls the `ccrz.ccLogicProductPricing.evaluateSubProdTermPricing` inner method for handling the dynamic kit's subscription pricing.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiProduct.PRODUCTPRICINGDATA`

`Map<String, Object>` of product pricing data, keyed by product ID, that results from the inner method calls.

SEE ALSO:

[Salesforce Help Create a Dynamic Kit](#)

## `ccrz.ccLogicProductPricing.evaluateAggregatePricing`

Evaluates criteria for pricing an aggregated product, including entitled price lists and base price lists. An aggregated product doesn't support sellers or subscriptions.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> evaluateAggregatePricing (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicProductPricing.evaluatePricing` method.

This method requires the following keys in the working map:

### `ccrz.ccApiPriceList.PRICELISTITEMS`

`Map<String, Map<String, List<Map<String, Object>>>>` where:

- The outer `Map` is keyed by a string that specifies a product ID.
- Each second-level `Map` is keyed by a string that specifies a type of price list item. For an aggregated product, only 'entitledPLIs' is supported.
- Each inner `List<Map<String, Object>>` is a list of entitled price list items.

### `ccrz.ccApiPriceList.PRICELISTS`

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

### `ccrz.ccApiProduct.EVALUATIONMODE`

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.

### 'basePriceListItems'

`Map<String, Object>` of `ccrz__E_PriceListItem__c` data, keyed by ID, that belong to the storefront's base price list.

### 'completeInputData'

`Map<String, Object>` that contains a copy of all the input keys and values passed to this method.

### 'product'

`ccrz__E_Product__c` to price.

## Logic Performed

This method completes the following actions:

1. For each component product included in an aggregated product's `compositeProductsS` related list, this method completes the following actions:
  - a. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from an entitled price list.
  - b. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from the storefront's base price list.
2. For the aggregated product itself, this method completes the following actions:

- a. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the aggregated product's price from an entitled price list.
- b. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the aggregated product's price from the storefront's base price list.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiProduct.PRODUCTPRICINGDATA`

`Map<String, Object>` of product pricing data, keyed by product ID, that results from the inner method calls.

SEE ALSO:

[Salesforce Help Create an Aggregated Product](#)

## `ccrz.ccLogicProductPricing.evaluateAssemblyPricing`

Evaluates criteria for pricing an assembly product, including entitled price lists and base price lists. An assembly product doesn't support sellers or subscriptions.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> evaluateAssemblyPricing (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicProductPricing.evaluatePricing` method.

This method requires the following keys in the working map:

### `ccrz.ccApiPriceList.PRICELISTITEMS`

`Map<String, Map<String, List<Map<String, Object>>>>` where:

- The outer `Map` is keyed by a string that specifies a product ID.



- Each second-level `Map` is keyed by a string that specifies a type of price list item. For an assembly product, only 'entitledPLIs' is supported.
- Each inner `List<Map<String, Object>>` is a list of entitled price list items.

**ccrz.ccApiPriceList.PRICELISTS**

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

**ccrz.ccApiProduct.EVALUATIONMODE**

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.

**'basePriceListItems'**

`Map<String, Object>` of `ccrz__E_PriceListItem__c` data, keyed by ID, that belong to the storefront's base price list.

**'completeInputData'**

`Map<String, Object>` that contains a copy of all the input keys and values passed to this method.

**'product'**

`ccrz__E_Product__c` to price.

## Logic Performed

This method completes the following actions:

1. For each component product included in an assembly product's `compositeProductsS` related list, this method completes the following actions:
  - a. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from an entitled price list.
  - b. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the component product's price from the storefront's base price list.
2. For the assembly product itself, this method completes the following actions:
  - a. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the assembly product's price from an entitled price list.
  - b. Calls the `ccrz.ccLogicProductPricing.determinePrice` inner method for setting the assembly product's price from the storefront's base price list.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**

Map<String, Object> of product pricing data, keyed by product ID, that results from the inner method calls.

SEE ALSO:

[Salesforce Help Create an Assembly Product](#)

**ccrz.ccLogicProductPricing.evaluateCustomProductPricing**

Provides an extension point for evaluating criteria for pricing a product type other than the default types.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> evaluateCustomProductPricing(Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> from the [ccrz.ccLogicProductPricing.evaluatePricing](#) method.

By default, this method doesn't evaluate any keys in the working map.

**Logic Performed**

By default, this method contains only a call to [ccrz.ccLogicProductPricing.evaluateBundlePricing](#).

**Outputs**

This method returns the working Map<String, Object>, and doesn't add or remove any keys.

**ccrz.ccLogicProductPricing.evaluateSellersPricing**

Evaluates criteria for pricing a standard product associated with a seller.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> evaluateSellersPricing (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicProductPricing.evaluateBasicPricing` method.

This method requires the following keys in the working map:

### **ccrz.ccApiPriceList.PRICELISTS**

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

### **ccrz.ccApiProduct.EVALUATIONMODE**

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.

### **'completeInputData'**

`Map<String, Object>` that contains a copy of all the input keys and values passed to `ccrz.ccLogicProductPricing.evaluateBasicPricing`.

### **'priceListItems'**

`List<Map<String, Object>>` of `ccrz__E_PriceListItem__c` data to evaluate. The price list items all apply to the same seller and the same product.

## Logic Performed

This method passes the input data to `ccrz.ccLogicProductPricing.determinePrice`, which returns the best price for the product from among the seller price list items.

## Outputs

This method returns the output from `ccrz.ccLogicProductPricing.determinePrice`.

SEE ALSO:

[ccrz\\_\\_E\\_Seller\\_\\_c](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccLogicProductPricing.evaluateSubProdTermPricing

Evaluates criteria for pricing a standard product, bundle, or dynamic kit associated with a subscription.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> evaluateSubProdTermPricing(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from any of the following methods:

- [ccrz.ccLogicProductPricing.evaluateBasicPricing](#)
- [ccrz.ccLogicProductPricing.evaluateBundlePricing](#)
- [ccrz.ccLogicProductPricing.evaluateDynamicKitPricing](#)

This method requires the following keys in the working map:

### **ccrz.ccApiPriceList.PRICELISTS**

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

### **ccrz.ccApiProduct.EVALUATIONMODE**

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.

**'completeInputData'**

Map<String, Object> that contains a copy of all the input keys and values passed in.

**'priceListItems'**

List<Map<String, Object>> of `ccrz__E_PriceListItem__c` data to evaluate.

**'subProdTerms'**

Map<String, Object> of `ccrz__E_SubProdTerm__c` data to evaluate.

## Logic Performed

This method completes the following actions:

1. Parses 'priceListItems' for recurring and upfront (non-recurring) subscription price data.
2. Calculates the minimum and maximum subscription price.
3. Evaluates any subscription modifiers that exist.
4. Calls `ccrz.ccLogicProductPricing.determinePrice`, which returns the best recurring and non-recurring price for the product from among the price list items.

## Outputs

This method returns the output from `ccrz.ccLogicProductPricing.determinePrice`, and sets the following keys:

**'returnPriceData'**

Map<String, Map<String, Object>> where:

- The outer map is keyed by `ccrz__E_SubProdTerm__c` Salesforce ID
- The inner map contains the following keys and values:

**'minMaxPrice'**

Map<String, Object> of the subscription's upfront (non-recurring) minimum and maximum price.

**'modifierSubscriptions'**

List<Map<String, Object>> of subscription modifier data.

**'nonRecurringPrice'**

Decimal that specifies the subscription's upfront price.

**'recurringPrice'**

Decimal that specifies the subscription's recurring price.

**'subProdTerms'**

Map<String, Object> of ccrz\_\_E\_SubProdTerm\_\_c data that was evaluated.

SEE ALSO:

[ccrz\\_\\_E\\_SubProdTerm\\_\\_c](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz.ccLogicProductPricing.evaluateKitProductSubscriptions

Evaluates criteria for pricing a kit associated with a subscription.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> evaluateKitProductSubscriptions(Map<String, Object>)
```

### Inputs (Required)

This method accepts the working Map<String, Object> from the [ccrz.ccLogicProductPricing.evaluateKitPricing](#) method.

This method requires the following keys in the working map:

**'compositesPricingData'**

Map<String, Object> that describes the prices of the kit's component products from an entitled price list.

**'priceListItems'**

List<Map<String, Object>> of ccrz\_\_E\_PriceListItem\_\_c data to evaluate.

**'productPrice'**

Map<String, Object> that describes the sum of the kit's component product entitled price list items.

**'subProdTerms'**

Map<String, Object> of ccrz\_\_E\_SubProdTerm\_\_c data to evaluate.

## Logic Performed

This method completes the following actions:

1. Parses 'priceListItems' for recurring and upfront (non-recurring) subscription price data.
2. Loops through price list items, and sets the recurring and non-recurring price for the kit.
3. Calculates the minimum and maximum subscription price.
4. Evaluates any subscription modifiers that exist.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### 'returnPriceData'

`Map<String, Map<String, Object>>` where:

- The outer map is keyed by `ccrz__E_SubProdTerm__c` Salesforce ID
- The inner map contains the following keys and values:

#### 'minMaxPrice'

`Map<String, Object>` of the subscription's upfront (non-recurring) minimum and maximum price.

#### 'modifierSubscriptions'

`List<Map<String, Object>>` of subscription modifier data.

#### 'nonRecurringPrice'

Decimal that specifies the subscription's upfront price.

#### 'recurringPrice'

Decimal that specifies the subscription's recurring price.

### 'subProdTerms'

`Map<String, Object>` of `ccrz__E_SubProdTerm__c` data that was evaluated.

SEE ALSO:

[Salesforce Help Create a Dynamic Kit](#)

[Salesforce Help Offer Subscriptions for Products on Your Storefront](#)

## ccrz.ccLogicProductPricing.determinePrice

Sets a price for any product when called from another evaluation method. This method is the primary logic for applying a price from an entitled price list, base price list, seller, or subscription. This method also evaluates any price list items associated with a contract pricing's price group, which take priority over an entitled price list.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> determinePrice (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from any of the following methods:

- `ccrz.ccLogicProductPricing.evaluateBasicPricing`
- `ccrz.ccLogicProductPricing.evaluateBundlePricing`
- `ccrz.ccLogicProductPricing.evaluateKitPricing`
- `ccrz.ccLogicProductPricing.evaluateDynamicKitPricing`
- `ccrz.ccLogicProductPricing.evaluateAggregatePricing`
- `ccrz.ccLogicProductPricing.evaluateAssemblyPricing`
- `ccrz.ccLogicProductPricing.evaluateSellersPricing`
- `ccrz.ccLogicProductPricing.evaluateSubProdTermPricing`

This method requires the following keys in the working map:

### `ccrz.ccApiPriceList.PRICELISTS`

`Map<String, Object>` of `ccrz__E_PriceList__c` data, keyed by ID.

### `ccrz.ccApiProduct.EVALUATIONMODE`

String that specifies one of the following constants, which references the [price list selection method for the buyer's account group](#):

Value	Usage
<code>ccrz.ccApiProduct.BESTPRICE</code> (default)	B2B Commerce for Visualforce evaluates all price lists assigned to the account group, and applies the lowest price available for each product.
<code>ccrz.ccApiProduct.SEQUENCE</code>	B2B Commerce for Visualforce evaluates the Sequence values on each assigned price list's Account Group Price List related record. For the Account Group Price List with the lowest Sequence value, B2B Commerce applies that price list's price.

### 'completeInputData'

`Map<String, Object>` that contains a copy of all the input keys and values passed in.

### 'priceListItems'

`List<Map<String, Object>>` of `ccrz__E_PriceListItem__c` data to evaluate.

### 'selectedPLIs'

`Set<String>` that's initially empty.



## Logic Performed

This method completes the following actions for each price list item in `'priceListItems'`:

1. Evaluates whether the price list item is associated with a price group:
  - When a price group exists, this method finds the value of the `sequence` field in the related `ccrz__E_PriceGroupPricelist__c` junction object.
  - Otherwise, this method finds the value of the `sequence` field in the related `ccrz__E_AccountGroupPricelist__c` junction object.
2. Checks `ccrz.ccApiProduct.EVALUATIONMODE` and, depending on the specified mode, builds a map of price data from the price list item that the method applied to the product.
3. Adds the price list item ID to `'selectedPLIs'`.
4. If the price list item defines any pricing tiers, this method calls into `ccrz.ccLogicProductPricing.processPricingTiersForPLI` for evaluating the tiers.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **'returnPriceData'**

`Map<String, Object>` that contains the following keys and values:

#### **'itemType'**

String that specifies the product type for the price list item.

#### **'price'**

Decimal that specifies the applied price from the price list item.

#### **'priceList'**

String that specifies the Salesforce ID of the `ccrz__E_PriceList__c` that the price list item belongs to.

#### **'seller'**

String that specifies the Salesforce ID of the `ccrz__E_Seller__c` associated with the price list item.

#### **'sfid'**

String that specifies the Salesforce ID of the `ccrz__E_PriceListItem__c` applied to the product.

### **'selectedPLIs'**

`Set<String>` that contains the Salesforce IDs of the applied price list items.

SEE ALSO:

[ccrz\\_\\_E\\_Pricelist\\_\\_c](#)

[ccrz\\_\\_E\\_PricelistItem\\_\\_c](#)

[Salesforce Help Host a Seller Marketplace on Your Storefront](#)

## ccrz.ccLogicProductPricing.processSelectedPLIs

Queries attributes or tiers pricing data for the price list items whose prices the `ccrz.ccLogicProductPricing.determinePrice` method applied.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processSelectedPLIs (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the inner methods of the `ccrz.ccLogicProductPricing.evaluatePricing` method complete.

This method requires the following keys in the working map:

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**  
`Map<String, Object>` of product pricing data.

**'selectedPLIs'**  
`Set<String>` that contains the Salesforce IDs of the applied price list items.

## Inputs (Optional)

If the working map includes the following key, this method evaluates it:

**ccrz.ccApiProduct.INCLUDE\_ATTRIBUTE\_PRICING**  
 Boolean

Value	Usage
<code>true</code>	Query for <code>ccrz__E_Attribute__c</code> records that are related to each product, and return the attribute pricing JSON applied to the selected price list items.
<code>false</code> (default)	Don't return any attribute pricing.

**ccrz.ccApiProduct.INCLUDE\_TIER\_PRICING**  
 Boolean

Value	Usage
<i>true</i>	Query tiered pricing JSON. For more information about tiered pricing, see <a href="https://help.salesforce.com/articleView?id=sf.b2b_commerce_tiered_pricing.htm">https://help.salesforce.com/articleView?id=sf.b2b_commerce_tiered_pricing.htm</a> .
<i>false</i> (default)	Don't return any tiered pricing.

**ccrz.ccApiProduct.PARSE\_ATTRIBUTE\_PRICING**

Boolean

Value	Usage
<i>true</i> (default)	Parse the attribute pricing JSON.
<i>false</i>	Skip attribute pricing.

**Logic Performed**

This method completes the following actions:

1. Passes 'selectedPLIs' as input for [ccrz.ccApiPriceList.fetchPriceListItems](#).
2. For each price list item in the `ccrz.ccApiPriceList.PRICELISTITEMS` output from `ccrz.ccApiPriceList.fetchPriceListItems`, this method completes the following actions:
  - a. Evaluates whether the price list item includes attributes pricing. If so, the method builds a `Map<String, Object>` that contains the following keys and values:
    - 'attr2Q'  
 Map<String, List<Object>> that describes parent and child attribute IDs. This method calls the `ccrz.ccLogicProductPricing.gatherAttributesToQuery` utility method for building this map.
    - 'attrMap'  
 Map<String, Object> that contains the deserialized attributes pricing JSON data.
  - b. Evaluates whether the price list item includes pricing tiers. If so, the method calls [ccrz.ccLogicProductPricing.processPricingTiersForPLI](#) for evaluating the tiers.

**Outputs**

This method returns the working `Map<String, Object>` after the inner methods complete, with the following key set:

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**

Map<String, Object> of product pricing data, updated with attributes or tiers pricing data.

This method also removes the following intermediate keys from the working map:

- 'selectedPLIs'

SEE ALSO:

[ccrz\\_\\_E\\_PricelistItem\\_\\_c](#)

[ccrz.ccApiPriceList.fetchPriceListItems](#)

## ccrz.ccLogicProductPricing.processPricingTiersForPLI

Evaluates product pricing data for an individual price list item and calculates the unit price.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processPricingTiersForPLI (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from any of the following methods:

- [ccrz.ccLogicProductPricing.determinePrice](#)
- [ccrz.ccLogicProductPricing.processSelectedPLIs](#)

This method requires the following keys in the working map:

**'completeInputData'**

`Map<String, Object>` that contains a copy of all the input keys and values passed in.

**'pli'**

`Map<String, Object>` that describes the `ccrz__E_PriceListItem__c` to evaluate.

**'productPrice'**

`Map<String, Object>` that contains the product pricing data to evaluate.

### Logic Performed

This method completes the following actions:

1. Deserializes the pricing tiers JSON data from the `pricingTiers` field in `'pli'` to a `Map<String, Object>`.
2. Parses the pricing tiers and calculates the unit price for the price list item.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**'productPrice'**

Map<String, Object> that contains the updated product pricing data with the unit price set.

SEE ALSO:

[Salesforce Help Adjust Product Prices Based on Tiered Quantities](#)

## ccrz.ccLogicProductPricing.processAttributes

Queries attributes data processed by the `ccrz.ccLogicProductPricing.processSelectedPLIs` method.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processAttributes (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working Map<String, Object> after the `ccrz.ccLogicProductPricing.processSelectedPLIs` method completes.

This method requires the following keys in the working map:

**attributesToQuery**

Map<String, List<Object>> that describes the parent and child attribute IDs to query. The `ccrz.ccLogicProductPricing.gatherAttributesToQuery` utility method, called from `ccrz.ccLogicProductPricing.processSelectedPLIs`, builds this map.

**ccrz.ccApiProduct.PARSE\_ATTRIBUTE\_PRICING**

Boolean

Value	Usage
<code>true</code> (default)	Parse the attribute pricing JSON. This method executes only when this value is <code>true</code> .
<code>false</code>	Skip attribute pricing.

**ccrz.ccApiProduct.PRODUCTPRICINGDATA**  
 Map<String, Object> of product pricing data.

## Logic Performed

This method completes the following actions:

1. Passes `attributesToQuery` as input for `ccrz.ccApiAttribute.fetch`.
2. Parses the `ccrz.ccApiPriceList.ATTRIBUTES` output from `ccrz.ccApiAttribute.fetch` and applies the attribute pricing to `ccrz.ccApiProduct.PRODUCTPRICINGDATA`.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApiAttribute.IDS**  
 List<String> of Salesforce IDs of attributes that were evaluated.

SEE ALSO:

[ccrz.ccApiAttribute.fetch](#)

## ccrz.ccLogicProductPricing.prepReturnData

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturnData (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working Map<String, Object> after the `ccrz.ccLogicProductPricing.processAttributes` method completes.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicProductUrlFor

Define the business logic for returning the friendly URLs for products.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Global API Caller

[ccrz.ccApiProduct.getUrlFor](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

#### [processInput](#)

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

#### [fetchProductCategories](#)

Queries for product categories and categories when the original input data already specifies

`ccrz.ccApiProduct.PRODUCTLIST`.

#### [fetchProducts](#)

Queries for products and categories when the original input data doesn't specify `ccrz.ccApiProduct.PRODUCTLIST`.

#### [fetchCategoryURLs](#)

Calls the `ccrz.ccApiCategory.getUrlFor` method for returning `ccrz__E_SiteIndex__c` records, which associate each category to its friendly URL.

#### [mapProductUrl](#)

Creates a map that associates the Salesforce ID of each product to its friendly URL.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

SEE ALSO:

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

## ccrz.ccLogicProductUrlFor.process

Overrides the base `ccrz.ccLogic.process` method, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiProduct.getUrlFor` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicProductUrlFor` class in the following order:

1. `processInput`
2. `fetchProductCategories` or `fetchProducts`
3. `fetchCategoryUrls`
4. `mapProductUrl`
5. `prepReturn`

### Outputs

This method returns the resulting `Map<String, Object>` after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.



## ccrz.ccLogicProductUrlFor.processInput

Copies the original input data to a new working map to pass through other downstream methods in the process chain. This method also parses the original input data for keys that downstream methods require, and verifies that appropriate values are set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicProductUrlFor.process` method.

This method checks that the input map specifies one of the following keys, and returns a `ccrz.ccApi.MissingInputException` if none of these keys are included:

**ccrz.ccApiProduct.PRODUCTIDLIST**

`List<String>` of Salesforce IDs for specific products whose friendly URLs you want to return.

**ccrz.ccApiProduct.PRODUCTLIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_Product__c` whose friendly URL you want to return.

**ccrz.ccApiProduct.PRODUCTSKULIST**

`List<String>` of specific product SKUs whose friendly URLs you want to return.

### Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiProduct.PRODUCTLOCALE**

String that specifies one of your [storefront's supported locales](#). The logic chain returns internationalized friendly URL values for the locale.

**ccrz.ccApiProduct.PRODUCTLOCALES**

`Set<String>` of your [storefront's supported locales](#). The logic chain returns internationalized friendly URL values for the locales.

**ccrz.ccApiProduct.STOREFRONT**

String that specifies the name of a storefront.

### Logic Performed

This method completes the following actions:

1. Looks for `ccrz.ccApiProduct.PRODUCTLIST`, and completes the following actions:
  - a. Adds each product ID to the `ccrz.ccApiProduct.PRODUCTIDLIST` output.
  - b. Sets `ccrz.ccApiProduct.PRODUCT_URL_FOR_QUERYSTRING` to a query of `ccrz__E_ProductCategory__c` records.
2. If `ccrz.ccApiProduct.PRODUCTLIST` isn't specified, the method looks for `ccrz.ccApiProduct.PRODUCTSKULIST`, and sets `ccrz.ccApiProduct.PRODUCT_URL_FOR_QUERYSTRING` to a query of `ccrz__E_Product__c` records.
3. If `ccrz.ccApiProduct.PRODUCTLIST` or `ccrz.ccApiProduct.PRODUCTIDLIST` aren't specified, the method looks for `ccrz.ccApiProduct.PRODUCTIDLIST`, and sets `ccrz.ccApiProduct.PRODUCT_URL_FOR_QUERYSTRING` to a query of `ccrz__E_Product__c` records.
4. Evaluates the optional inputs and sets fallback values, if necessary.

## Outputs

This method returns a `Map<String, Object>`, which is a copy of the original input data that other downstream methods can process. This map is known as the working map. This method also adds the following keys to the working map.

### **ccrz.ccApiCategory.CATEGORYLOCALE**

String based on the value of the `ccrz.ccApiProduct.PRODUCTLOCALE` input.



**Note:** If the input data doesn't specify `ccrz.ccApiProduct.PRODUCTLOCALE`, the method falls back to `ccrz.cc_CallContext.userLocale`.

### **ccrz.ccApiCategory.CATEGORYLOCALES**

String based on the value of the `ccrz.ccApiProduct.PRODUCTLOCALES` input.

### **ccrz.ccApiProduct.PRODUCTIDLIST**

`List<String>` of Salesforce IDs for specific products whose friendly URLs you want to return.

### **ccrz.ccApiProduct.PRODUCT\_URL\_FOR\_QUERYSTRING**

String that specifies a query of product or product category records.

### **ccrz.ccApiProduct.STOREFRONT**

String based on the value of the `ccrz.ccApiProduct.STOREFRONT` input.



**Note:** If the input data doesn't specify this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

## ccrz.ccLogicProductUrlFor.fetchProductCategories

Queries for product categories and categories when the original input data already specifies `ccrz.ccApiProduct.PRODUCTLIST`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> fetchProductCategories (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working map `Map<String, Object>` after the `ccrz.ccLogicProductUrlFor.processInput` method completes.

 **Note:** This method executes only when the working map includes the `ccrz.ccApiProduct.PRODUCTLIST` input.

This method requires the following keys in the working map:

**ccrz.ccApiProduct.PRODUCT\_URL\_FOR\_QUERYSTRING**  
String that specifies a query of product or product category records.

**ccrz.ccApiProduct.PRODUCTLOCALE**  
String that specifies one of your [storefront's supported locales](#). The logic chain returns internationalized friendly URL values for the locale.

## Logic Performed

This method completes the following actions:

1. Executes a query of `ccrz__E_ProductCategory__c` records using `ccrz.ccApiProduct.PRODUCT_URL_FOR_QUERYSTRING`.
2. Parses each queried `ccrz__E_ProductCategory__c` record for the referenced `ccrz__E_Category__c`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiCategory.CATEGORYIDLIST**  
`List<String>` of Salesforce IDs of `ccrz__E_Category__c` records related to each product.

**ccrz.ccLogicProductUrlFor.PRODUCT\_CATEGORIES**  
`Map<String, List<ccrz__E_ProductCategory__c>>`, keyed by product ID.

SEE ALSO:

[ccrz\\_\\_E\\_Category\\_\\_c](#)

[ccrz\\_\\_E\\_ProductCategory\\_\\_c](#)

## ccrz.ccLogicProductUrlFor.fetchProducts

Queries for products and categories when the original input data doesn't specify `ccrz.ccApiProduct.PRODUCTLIST`.

## Compatibility

This reference applies to:


Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> fetchProducts (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working map `Map<String, Object>` after the `ccrz.ccLogicProductUrlFor.processInput` method completes.

 **Note:** This method executes only when the working map doesn't include the `ccrz.ccApiProduct.PRODUCTLIST` input.

This method requires the following keys in the working map:

**ccrz.ccApiProduct.PRODUCT\_URL\_FOR\_QUERYSTRING**  
String that specifies a query of product or product category records.

## Inputs (Optional)

If the working map includes the following keys, this method evaluates them:

**ccrz.ccApiProduct.PRODUCTLOCALE**  
String that specifies one of your [storefront's supported locales](#). The logic chain returns internationalized friendly URL values for the locale.

**ccrz.ccApiProduct.PRODUCTLOCALES**  
`Set<String>` of your [storefront's supported locales](#). The logic chain returns internationalized friendly URL values for the locales.

## Logic Performed

This method completes the following actions:

1. Evaluates `ccrz.ccApiProduct.PRODUCTLOCALE` or `ccrz.ccApiProduct.PRODUCTLOCALES` into a single input, `ccrz.ccApiProduct.PRODUCTLOCALES`.
2. Executes a query of `ccrz__E_Product__c` records using `ccrz.ccApiProduct.PRODUCT_URL_FOR_QUERYSTRING`.
3. Parses each queried product's list of related `ccrz__E_ProductCategory__c` records for the referenced `ccrz__E_Category__c`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiCategory.CATEGORYIDLIST**

List<String> of Salesforce IDs of `ccrz__E_Category__c` records related to each product.

**ccrz.ccApiProduct.PRODUCTLIST**

List<`ccrz__E_Product__c`> of the products whose friendly URLs you want to return.

SEE ALSO:

[ccrz\\_\\_E\\_Category\\_\\_c](#)

[ccrz\\_\\_E\\_Product\\_\\_c](#)

[ccrz\\_\\_E\\_ProductCategory\\_\\_c](#)

## ccrz.ccLogicProductUrlFor.fetchCategoryURLs

Calls the `ccrz.ccApiCategory.getUrlFor` method for returning `ccrz__E_SiteIndex__c` records, which associate each category to its friendly URL.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

### Signature

```
global virtual Map<String, Object> fetchCategoryURLs (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working map `Map<String, Object>` after the `ccrz.ccLogicProductUrlFor.fetchProductCategories` or `ccrz.ccLogicProductUrlFor.fetchProducts` method completes.

This method requires the following key in the working map:

**ccrz.ccApiCategory.CATEGORYIDLIST**

List<String> of Salesforce IDs of `ccrz__E_Category__c` records.

This method also requires one of the following keys in the working map:

**ccrz.ccApiCategory.CATEGORYLOCALE**

String that specifies one of your storefront's supported locales.

**ccrz.ccApiCategory.CATEGORYLOCALES**

Set<String> of your storefront's supported locales.

## Logic Performed

This method completes the following actions:

1. If the input data includes `ccrz.ccApiCategory.CATEGORYLOCALE`, the method returns the `ccrz.ccApiCategory.CATEGORYURLMAP` or `ccrz.ccApiCategory.SITEINDICES` output from [ccrz.ccApiCategory.getUrlFor](#).
2. If the input data includes `ccrz.ccApiCategory.CATEGORYLOCALES`, the method returns the `ccrz.ccApiCategory.LOCALECATEGORYURLMAP` or `ccrz.ccApiCategory.LOCALESITEINDICES` output from [ccrz.ccApiCategory.getUrlFor](#).

## Outputs

This method returns the working `Map<String, Object>`, updated with the [ccrz.ccApiCategory.getUrlFor](#) output:

### **ccrz.ccApiCategory.CATEGORYURLMAP**

`Map<String, Object>` for a single locale, where each key is a category ID and each value is the category's friendly URL.

### **ccrz.ccApiCategory.LOCALECATEGORYURLMAP**

`Map<String, Map<String, Object>>` for multiple locales, where:

- The outer map is keyed by category ID.
- Each inner map is keyed by locale.
- Each inner map's value is the category's friendly URL for the locale.

### **ccrz.ccApiCategory.LOCALESITEINDICES**

`Map<String, Map<String, Object>>` for multiple locales, where:

- The outer map is keyed by category ID.
- Each inner map is keyed by locale.
- Each inner map's value is the category's corresponding `ccrz__E_SiteIndex__c` record for the locale.

### **ccrz.ccApiCategory.SITEINDICES**

`Map<String, Object>` for a single locale, where each key is a category ID and each value is the corresponding `ccrz__E_SiteIndex__c` record.

SEE ALSO:

[ccrz\\_\\_E\\_SiteIndex\\_\\_c](#)

[ccrz.ccApiCategory.getUrlFor](#)

[ccrz.ccLogicCategoryUrlFor.fetchSiteIndices](#)

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

## ccrz.ccLogicProductUrlFor.mapProductUrl

Creates a map that associates the Salesforce ID of each product to its friendly URL.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> mapProductUrl (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working map `Map<String, Object>` after the `ccrz.ccLogicProductUrlFor.fetchCategoryURLs` method completes.

## Logic Performed

This method completes the following actions:

1. If the original input data included `ccrz.ccApiProduct.PRODUCTLIST`, the `ccrz.ccLogicProductUrlFor.fetchProductCategories` method executed, and the working map includes `ccrz.ccLogicProductUrlFor.PRODUCT_CATEGORIES`, this method calls the private `mapProductUrlByProductCategory` method, which:
  - a. Evaluates the following keys:
    - `ccrz.ccApiCategory.CATEGORYURLMAP`
    - `ccrz.ccApiCategory.SITEINDICES`
    - `ccrz.ccApiProduct.PRODUCTLIST`
    - `ccrz.ccLogicProductUrlFor.PRODUCT_CATEGORIES`
  - b. Returns `ccrz.ccApiProduct.SITEINDEX_PRODUCT_MAP` and `ccrz.ccApiProduct.PRODUCTURLMAP`
2. Otherwise, the `ccrz.ccLogicProductUrlFor.fetchProducts` method executed, and this method calls the private `mapProductUrlByProduct` method, which:
  - a. Evaluates the following keys:
    - `ccrz.ccApiCategory.LOCALECATEGORYURLMAP`
    - `ccrz.ccApiCategory.LOCALESITEINDICES`
    - `ccrz.ccApiProduct.PRODUCTLIST`
    - `ccrz.ccApiProduct.PRODUCTSKULIST`
    - `ccrz.ccApiProduct.PRODUCTLOCALES`
  - b. Returns `ccrz.ccApiProduct.LOCALE_SITEINDEX_PRODUCT_MAP` and `ccrz.ccApiProduct.LOCALE_PRODUCTURLMAP`

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiProduct.LOCALE_SITEINDEX_PRODUCT_MAP`

`Map<String, Object>` for multiple locales, where:

- Each key is a product ID. If the input data includes `ccrz.ccApiProduct.PRODUCTSKULIST`, each key is a SKU instead.
- each value is the `ccrz__E_SiteIndex__c` record that corresponds to the product's category for a specific locale.

### `ccrz.ccApiProduct.LOCALE_PRODUCTURLMAP`

`Map<String, Map<String, Object>>` for multiple locales, where:

- The outer map is keyed by product ID. If the input data includes `ccrz.ccApiProduct.PRODUCTSKULIST`, the outer map is keyed by SKU instead.
- Each inner map is keyed by locale.
- Each inner map's value is the product's friendly URL for the locale.

### `ccrz.ccApiProduct.PRODUCTURLMAP`

`Map<String, Object>` for a single locale, where each key is a product ID and each value is the product's friendly URL.

### `ccrz.ccApiProduct.SITEINDEX_PRODUCT_MAP`

`Map<String, Object>` for a single locale, where each key is a product ID and each value is the `ccrz__E_SiteIndex__c` record that corresponds to the product's category.

SEE ALSO:

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

## `ccrz.ccLogicProductUrlFor.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```



## Inputs (Required)

This method accepts the working map `Map<String, Object>` after the `ccrz.ccLogicProductUrlFor.mapProductUrl` method completes.

## Outputs

This method removes the following intermediate keys from the working map, and returns the final `Map<String, Object>`:

- `ccrz.ccApiCategory.CATEGORYIDLIST`
- `ccrz.ccApiCategory.CATEGORYLOCALE`
- `ccrz.ccApiCategory.CATEGORYLOCALES`
- `ccrz.ccApiCategory.CATEGORYURLMAP`
- `ccrz.ccApiCategory.LOCALECATEGORYURLMAP`
- `ccrz.ccApiCategory.LOCALESITEINDICES`
- `ccrz.ccApiProduct.PRODUCT_URL_FOR_QUERYSTRING`
- `ccrz.ccApiProduct.PRODUCTIDLIST`
- `ccrz.ccApiProduct.STOREFRONT`
- `ccrz.ccLogic.ORIGINAL_INPUT`
- `ccrz.ccLogicProductUrlFor.PRODUCT_CATEGORIES`

## ccrz.ccLogicSICreate

Define the business logic for creating site index records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiSiteIndex.create`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [processCategoryData](#)

Returns a data structure of the parent categories for the category IDs that you specified for the `ccrz.ccApiSiteIndex.create` method. The method calls the `ccrz.ccLogicSICreate.fetchCategories` and `ccrz.ccLogicSICreate.processCategoryNodes` methods in a loop to traverse up the category tree to the storefront's root category.

#### [fetchCategories](#)

Returns the category records and up to five levels of parent category relationships for each category in the `ccrz.ccApiSiteIndex.CATEGORY_IDS` set.

#### [processCategoryNodes](#)

Determines whether more parent categories—up to five levels of parent hierarchy—exist for the current node in the category tree. If more parent categories exist, those parent categories are added to the `ccrz.ccApiSiteIndex.CATEGORY_IDS` set for further loop processing.

#### [fetchLocalizedData](#)

Queries each category's internationalized records.

#### [createSiteIndices](#)

Creates the site index sObjects, including assigning the category's friendly URL and creating the breadcrumb path.

#### [insertSiteIndices](#)

Executes the DML `insert` operation for the created site index records, which assigns IDs to each record.

#### [refetch](#)

Call the `ccrz.ccApiSiteIndex.fetch` method for fetching the site index records after they're created and inserted.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## `ccrz.ccLogicSICreate.process`

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiSiteIndex.create` method.


## Logic Performed

This method invokes other methods of the `ccrz.ccLogicSICreate` class in the following order:

1. `processInput`
2. `processCategoryData`, which calls `fetchCategories` and `processCategoryNodes` in a loop
3. `fetchLocalizedData`
4. `createSiteIndices`
5. `processSiteIndices`
6. `insertSiteIndices`
7. `refetch`
8. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## `ccrz.ccLogicSICreate.processInput`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicSICreate.process` method.

This method checks that the input map specifies the following keys, and returns a `ccrz.ccApi.MissingInputException` if any key is missing:

### `ccrz.ccApiSiteIndex.CATEGORY_IDS`

`Set<String>` of category IDs to create site index records for.

### `ccrz.ccApiSiteIndex.LOCALES`

`Set<String>` of locales to create site index records for. Typically, you want to create site index records for all the locales that your storefront supports.

### `ccrz.ccApiSiteIndex.ROOT_CATEGORY_ID`

String that specifies the ID of the [storefront's root category](#).



**Note:** If this key isn't specified, the method doesn't return an exception. But, this key ensures that the created site index records specify the correct storefront and reflect the storefront's category tree.

### `ccrz.ccApiSiteIndex.STOREFRONT`

String that specifies the name of the storefront to create site index records for.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccLogic.ORIGINAL_INPUT`

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicSICreate.prepReturn` method removes this intermediate key from the working map.

SEE ALSO:

[Salesforce Help Organize Products into Categories for B2B Commerce for Visualforce](#)

## ccrz.ccLogicSICreate.processCategoryData

Returns a data structure of the parent categories for the category IDs that you specified for the `ccrz.ccApiSiteIndex.create` method. The method calls the `ccrz.ccLogicSICreate.fetchCategories` and `ccrz.ccLogicSICreate.processCategoryNodes` methods in a loop to traverse up the category tree to the storefront's root category.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processCategoryData (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSICreate.processInput` method.

This method requires the following key in the working map:

#### **ccrz.ccApiSiteIndex.CATEGORY\_IDS**


`Set<String>` of category IDs. Typically, these categories are *leaves* in the category tree: categories with no child categories. The method calls the `ccrz.ccLogicSICreate.fetchCategories` and `ccrz.ccLogicSICreate.processCategoryNodes` methods in a loop to traverse up the category tree to the storefront's root category.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

#### **'processedCategoryNodes'**

`Map<String, ccrz__E_Category__c>` where each entry contains a category in the category tree. By default, each entry can specify up to five levels of parent categories.

 **Note:** What happens if the `ccrz.ccApiSiteIndex.CATEGORY_IDS` key includes a category ID that's more than five levels deep in the category tree? In this case, the default `ccrz.ccLogicSICreate` class can't create its site index record successfully.

SEE ALSO:

[Salesforce Help Organize Products into Categories for B2B Commerce for Visualforce](#)

## ccrz.ccLogicSICreate.fetchCategories

Returns the category records and up to five levels of parent category relationships for each category in the `ccrz.ccApiSiteIndex.CATEGORY_IDS` set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchCategories (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSICreate.processCategoryData` method.

This method requires the following key in the working map:

**ccrz.ccApiSiteIndex.CATEGORY\_IDS**

`Set<String>` of category IDs, either originally passed in to the `ccrz.ccApiSiteIndex.create` method or returned from the loop process with the `ccrz.ccLogicSICreate.processCategoryNodes` method.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**'categoryNodes'**

Map<String, ccrz\_\_E\_Category\_\_c> where each entry contains a category (or *node*) in the category tree, keyed by category ID. By default, each entry can specify up to five levels of parent categories.

SEE ALSO:

[Salesforce Help Organize Products into Categories for B2B Commerce for Visualforce](#)

## ccrz.ccLogicSICreate.processCategoryNodes

Determines whether more parent categories—up to five levels of parent hierarchy—exist for the current node in the category tree. If more parent categories exist, those parent categories are added to the `ccrz.ccApiSiteIndex.CATEGORY_IDS` set for further loop processing.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processCategoryNodes (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working Map<String, Object> from the `ccrz.ccLogicSICreate.fetchCategories` method.

This method requires the following key in the working map:

**'categoryNodes'**

Map<String, ccrz\_\_E\_Category\_\_c> where each entry contains a category (or *node*) in the category tree, keyed by category ID. By default, each entry can specify up to five levels of parent categories.

### Outputs

This method returns the working Map<String, Object>, and sets the following keys:

**ccrz.ccApiSiteIndex.CATEGORY\_IDS**

Set<String> of the available parent category IDs to continue processing with the `ccrz.ccLogicSICreate.fetchCategories` method. When the current node being processed no longer has any parent

category, the method determines that the loop process has reached the storefront's root category. At that point, the method removes the `ccrz.ccApiSiteIndex.CATEGORY_IDS` key from the working map.

**'completeCategoryIdSet'**

`Set<String>` of all category IDs and their parent categories. Each iteration of the `ccrz.ccLogicSICreate.processCategoryNodes` method updates this set.

SEE ALSO:

[Salesforce Help Organize Products into Categories for B2B Commerce for Visualforce](#)

## ccrz.ccLogicSICreate.fetchLocalizedData

Queries each category's internationalized records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchLocalizedData (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSICreate.processCategoryData` method.

This method requires the following keys in the working map:

**ccrz.ccApiSiteIndex.LOCALES**

`Set<String>` of locales to query internationalized category data for. Typically, you want to create site index records for all the locales that your storefront supports.

**'completeCategoryIdSet'**

`Set<String>` of all category IDs and their parent categories. The method queries `ccrz__E_CategoryI18N__c` records that correspond to these category IDs.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:



**'localizedCategoryData'**

Map<String, Map<String, ccrz\_\_E\_CategoryI18N\_\_c>> where each entry contains an internationalized category record, keyed by locale and category ID:

```
<Locale<Category ID, ccrz__E_CategoryI18N__c>>
```

## SEE ALSO:

[Salesforce Help Localize Categories for an Internationalized Storefront](#)

[Salesforce Help Translate Storefront Text for Buyers in Different Locales](#)

**ccrz.ccLogicSICreate.createSiteIndices**

Creates the site index sObjects, including assigning the category's friendly URL and creating the breadcrumb path.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> createSiteIndices (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> from the `ccrz.ccLogicSICreate.fetchLocalizedData` method.

This method requires the following keys in the working map:

**ccrz.ccApiSiteIndex.LOCALES**

Set<String> of locales to create site index records for. Typically, you want to create site index records for all the locales that your storefront supports.

**ccrz.ccApiSiteIndex.ROOT\_CATEGORY\_ID**

String that specifies the ID of the [storefront's root category](#).

**ccrz.ccApiSiteIndex.STOREFRONT**

String that specifies the name of the storefront to create site index records for.

**'localizedCategoryData'**

Map<String, Map<String, ccrz\_\_E\_CategoryI18N\_\_c>> where each entry contains an internationalized category record, keyed by locale and category ID:

```
<Locale<Category ID, ccrz__E_CategoryI18N__c>>
```

The `ccrz.ccLogicSICreate.fetchLocalizedData` method returns this key.

**'processedCategoryNodes'**

Map<String, ccrz\_\_E\_Category\_\_c> where each entry contains a category in the category tree, keyed by category ID. By default, each entry can specify up to five levels of parent categories. The

`ccrz.ccLogicSICreate.processCategoryData` method returns this key.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApiSiteIndex.SITE\_INDICES**

Map<String, ccrz\_\_E\_SiteIndex\_\_c> of the created site index sObjects, ready to insert with the `ccrz.ccLogicSICreate.insertSiteIndices` method.

**Note:**

- Each new site index record doesn't have an ID until inserted.
- Each new site index record's locale depends on the locales that your storefront supports and the locales that you've created internationalized category records for. If necessary, this method [assigns a fallback locale to a site index record instead of a full locale](#).

## SEE ALSO:

[Salesforce Help Localize Categories for an Internationalized Storefront](#)

[Salesforce Help Translate Storefront Text for Buyers in Different Locales](#)

## ccrz.ccLogicSICreate.insertSiteIndices

Executes the DML `insert` operation for the created site index records, which assigns IDs to each record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> insertSiteIndices (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSICreate.createSiteIndices` method.

This method requires the following key in the working map:

### `ccrz.ccApiSiteIndex.SITE_INDICES`

`Map<String, ccrz__E_SiteIndex__c>` of the site index sObjects to insert. The `ccrz.ccLogicSICreate.createSiteIndices` method returns this key.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiSiteIndex.CATEGORY_IDS`

`Set<String>` of category IDs that site index records were created for.

### `ccrz.ccApiSiteIndex.IDS`

`Set<String>` of IDs of the new `ccrz__E_SiteIndex__c` records.

SEE ALSO:

[Inserting and Updating Records](#)

## `ccrz.ccLogicSICreate.refetch`

Call the `ccrz.ccApiSiteIndex.fetch` method for fetching the site index records after they're created and inserted.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> refetch (Map<String, Object>)
```

## Inputs (Required)


This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSICreate.insertSiteIndices` method.

This method requires the following key in the working map:

### `ccrz.ccApi.SIZING`

`Map<String, Object>` that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiSiteIndex.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApiSiteIndex.SITE_INDICES`

`Map<String, Object>` of `ccrz__E_SiteIndex__c` records that the `ccrz.ccApiSiteIndex.fetch` method returns.

SEE ALSO:

[ccrz.ccApiSiteIndex.fetch](#)

## `ccrz.ccLogicSICreate.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSILogicURL.refetch` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicSILogicURL

Define the business logic for returning the category or product default URL that corresponds to a specified friendly URL.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiSiteIndex.mapUrl`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [processUrl](#)

Parses a specified friendly URL into separate pieces of data.

### [fetchSiteIndices](#)

Calls the `ccrz.ccApiSiteIndex.fetch` method to query site index records for matches against the hashed root and partial URL values.

### [processSiteIndices](#)

Evaluates the hashed URL portions, and attempts to match the URL with a category's default URL used on the Product List page.

### [fetchProducts](#)

Evaluates product category records and the SEO ID field on product records to find a product whose Product Detail page URL corresponds to the hashed URL portions. This method executes only when the `ccrz.ccLogicSMapUrl.processSiteIndices` method can't exactly match the working map's friendly URL with a category URL.

### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicSMapUrl.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiSiteIndex.mapUrl` method.

## Logic Performed


This method invokes other methods of the `ccrz.ccLogicSMapUrl` class in the following order:

1. `processInput`
2. `processUrl`

3. `fetchSiteIndices`
4. `processSiteIndices`
5. `fetchProducts`
6. `prepReturn`

## Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## `ccrz.ccLogicSISMapUrl.processInput`

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicSISMapUrl.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### `ccrz.ccApiSiteIndex.FULL_URL`

String that specifies the friendly URL for a product or category whose default URL you want to return. This value can be a URL that the `ccrz.ccApiProduct.getUrlFor` or `ccrz.ccApiCategory.getUrlFor` method returns.

## Inputs (Optional)

This method can also process the following keys, if specified, in the `Map<String, Object>` from the `ccrz.ccLogicSISMapUrl.process` method.

**ccrz.ccApiSiteIndex.STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.ccApiSiteIndex.URL\_LOCALE**

String that specifies one of your [storefront's supported locales](#).

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userLocale`.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicSMapUrl.prepReturn` method removes this intermediate key from the working map.

### Example: Example

Prefix the `ccrz.ccApiSiteIndex.FULL_URL` with the user's locale.

```
global virtual with sharing class customSiteIndexMapUrlLogic extends ccrz.ccLogicSMapUrl
{
    global virtual override Map<String, Object> processInput(Map<String, Object>
) {
        Map<String, Object> returnData = super.processInput(inputData);
        String fullUrl = (String) inputData.get(ccrz.ccApiSiteIndex.FULL_URL);
        String locale = ccrz.cc_CallContext.userLocale;
        fullUrl = fullUrl.removeStart('/') + locale;
        returnData.put(ccrz.ccApiSiteIndex.FULL_URL, fullUrl)
        return returnData;
    }
}
```

SEE ALSO:

[ccrz.ccApiCategory.getUrlFor](#)

[ccrz.ccApiProduct.getUrlFor](#)

## ccrz.ccLogicSMapUrl.processUrl

Parses a specified friendly URL into separate pieces of data.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processUrl (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSIMapUrl.process` method.

This method requires the following key in the working map:

### **ccrz.ccApiSiteIndex.FULL\_URL**

String that specifies the friendly URL for a product or category whose default URL you want to return. This value can be a URL that the `ccrz.ccApiProduct.getUrlFor` or `ccrz.ccApiCategory.getUrlFor` method returns.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### **ccrz.ccApiSiteIndex.URL\_HASH\_LIST**

`List<String>` of the `'rootUrl'` and `'partialUrl'` values, passed through the Apex `String.hashCode` method.

### **ccrz.ccApiSiteIndex.URL\_PARAMS**

String that specifies any URL parameters that appeared after a question mark character (?) in the `ccrz.ccApiSiteIndex.FULL_URL` input value.

### **'checkProductSEOId'**

String that matches the final portion of the path from `'rootUrl'`. For example, if `'rootUrl'` is `/Coffee/Medium/MediumRoast`, the `'checkProductSEOId'` portion is `/MediumRoast`.

### **'partialUrl'**

String that matches `'rootUrl'` but excludes the final portion of the path. For example, if `'rootUrl'` is `/Coffee/Medium/MediumRoast`, the `'partialUrl'` portion is `/Coffee/Medium`. The remaining portion of the path, such as `/MediumRoast`, is the value of `'checkProductSEOId'`.

**'rootUrl'**

String that specifies the portion of the `ccrz.ccApiSiteIndex.FULL_URL` value that precedes a question mark character (?). Any values that follow the question mark character are considered URL parameters and are included in the `ccrz.ccApiSiteIndex.URL_PARAMS` key.

## SEE ALSO:

[ccrz.ccApiCategory.getUrlFor](#)

[ccrz.ccApiProduct.getUrlFor](#)

[Salesforce Help Specify SEO Metadata for Products and Categories](#)

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

## ccrz.ccLogicSMapUrl.fetchSiteIndices

Calls the `ccrz.ccApiSiteIndex.fetch` method to query site index records for matches against the hashed root and partial URL values.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchSiteIndices (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicSMapUrl.processUrl` method.

This method requires the following keys in the working map:

**ccrz.ccApiSiteIndex.URL\_HASH\_LIST**

`List<String>` of the 'rootUrl' and 'partialUrl' values returned from, passed through the Apex `String.hashCode` method. The `ccrz.ccLogicSMapUrl.processUrl` method returns this key.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApiSiteIndex.SITE\_INDICES**

Map<String, ccrz\_\_E\_SiteIndex\_\_c> of site index records, keyed by ID, that match the query.

SEE ALSO:

[ccrz.ccApiSiteIndex.fetch](#)

**ccrz.ccLogicSIMapUrl.processSiteIndices**

Evaluates the hashed URL portions, and attempts to match the URL with a category's default URL used on the Product List page.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> processSiteIndices (Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working Map<String, Object> from the [ccrz.ccLogicSIMapUrl.fetchSiteIndices](#) method.

This method requires the following keys in the working map:

**ccrz.ccApiSiteIndex.SITE\_INDICES**

Map<String, ccrz\_\_E\_SiteIndex\_\_c> of site index records, keyed by ID, to evaluate. The [ccrz.ccLogicSIMapUrl.fetchSiteIndices](#) method returns this key.

**ccrz.ccApiSiteIndex.URL\_LOCALE**

String that specifies the storefront locale to evaluate URLs against.

**'rootUrlHash'**

String that specifies the friendly URL's 'rootUrl' value, passed through the Apex [String.hashCode](#) method.

**Inputs (Optional)**

This method can also process the following keys, if specified, in the working Map<String, Object>.

**'partialUrlHash'**

String that specifies the friendly URL's 'partialUrl' value, passed through the Apex [String.hashCode](#) method.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

### `ccrz.ccApiSiteIndex.MAPPED_URL`

String that specifies the category default URL that corresponds to the specified friendly URL. This key is returned when `'rootUrlHash'` matches a queried site index record, which indicates that the friendly URL corresponds to a category.

### `'checkProductCategory'`

String that specifies a category ID. This key is returned when `'partialUrlHash'` matches on the queried site index records. Because the URL only partially matched, the friendly URL doesn't definitively correspond to a category or product. The `ccrz.ccLogicSMapUrl.fetchProducts` method attempts to match the friendly URL with a product by querying `ccrz__E_ProductCategory__c` records for this category ID.

SEE ALSO:

[ccrz\\_\\_E\\_ProductCategory\\_\\_c](#)

[Salesforce Help Specify SEO Metadata for Products and Categories](#)

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

## `ccrz.ccLogicSMapUrl.fetchProducts`

Evaluates product category records and the SEO ID field on product records to find a product whose Product Detail page URL corresponds to the hashed URL portions. This method executes only when the `ccrz.ccLogicSMapUrl.processSiteIndices` method can't exactly match the working map's friendly URL with a category URL.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchProducts (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSMapUrl.processSiteIndices` method.

This method requires the following keys in the working map:

**'checkProductCategory'**

String that specifies a category ID to match against `ccrz__E_ProductCategory__c`. The `ccrz.ccLogicSMapUrl.processSiteIndices` method returns this key when 'partialUrlHash' matches on the queried site index records. Because the URL only partially matched, the friendly URL doesn't definitively correspond to a category or product.

**'checkProductSEOID'**

String that matches the final portion of the path from 'rootUrl', which the method tries to match against the SEO ID field on `ccrz__E_Product__c` records. For example, if 'rootUrl' is /Coffee/Medium/MediumRoast, the 'checkProductSEOID' portion is /MediumRoast. The `ccrz.ccLogicSMapUrl.processUrl` method returns this key.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApiSiteIndex.MAPPED\_URL**

String that specifies the product default URL that corresponds to the specified friendly URL.

SEE ALSO:

[ccrz\\_\\_E\\_Product\\_\\_c](#)

[ccrz\\_\\_E\\_ProductCategory\\_\\_c](#)

[Salesforce Help Specify SEO Metadata for Products and Categories](#)

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

## ccrz.ccLogicSMapUrl.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSIMapUrl.fetchProducts` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicSIRemove

Define the business logic for deleting site index records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiSiteIndex.remove`

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [fetchSiteIndices](#)

Calls the `ccrz.ccApiSiteIndex.fetch` method to return the site index records that you want to delete.

#### [deleteSiteIndices](#)

Executes the DML `delete` operation for the site index records.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicSIRemove.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiSiteIndex.remove` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicSIRemove` class in the following order:

1. `processInput`
2. `fetchSiteIndices`
3. `deleteSiteIndices`
4. `prepReturn`

### Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicSIRemove.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicSIRemove.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

#### **ccrz.ccApiSiteIndex.IDS**

`List<String>` of site index IDs to remove.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

#### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicSIRemove.prepReturn` method removes this intermediate key from the working map.

## ccrz.ccLogicSIRemove.fetchSiteIndices

Calls the `ccrz.ccApiSiteIndex.fetch` method to return the site index records that you want to delete.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchSiteIndices (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSIRemove.processInput` method.

This method requires the following key in the working map:

### **ccrz.ccApiSiteIndex.IDS**

`List<String>` of site index IDs to fetch.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccApiSiteIndex.SITE\_INDICES**

`Map<String, ccrz__E_SiteIndex__c>` of site index records, keyed by ID, that match the query.

SEE ALSO:

[ccrz.ccApiSiteIndex.fetch](#)

## ccrz.ccLogicSIRemove.deleteSiteIndices

Executes the DML `delete` operation for the site index records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> deleteSiteIndices(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSIRemove.fetchSiteIndices` method.

This method requires the following key in the working map:

**ccrz.ccApiSiteIndex.SITE\_INDICES**

`Map<String, ccrz__E_SiteIndex__c>` of site index records, keyed by ID, to delete.

## Outputs

This method returns the working `Map<String, Object>`, and doesn't add or remove any keys.

SEE ALSO:

[Deleting Records](#)

## ccrz.ccLogicSIRemove.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn(Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicSIRemove.deleteSiteIndices` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicStoredPaymentCreate

Define the business logic for creating a stored payment record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

[ccrz.ccApiStoredPayment.create](#)

## Methods

### IN THIS SECTION:

#### [process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

#### [processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

#### [decorateStoredPayment](#)

Evaluates the input fields and values and creates a corresponding `ccrz__E_StoredPayment__c` record, ready to insert.

#### [insertStoredPayment](#)

Executes the DML `insert` operation for the created stored payment record, which assigns an ID to the record.

#### [refetch](#)

Calls the `ccrz.ccApiStoredPayment.fetch` method for fetching the stored payment record after it's created and inserted.

#### [prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicStoredPaymentCreate.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiStoredPayment.create` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicStoredPaymentCreate` class in the following order:

1. `processInput`
2. `decorateStoredPayment`
3. `insertStoredPayment`
4. `refetch`
5. `prepReturn`

### Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicStoredPaymentCreate.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicStoredPaymentCreate.process` method.

This method checks that the input map specifies the following key, and returns a `ccrz.ccApi.MissingInputException` if the key is missing:

### **ccrz.ccApiStoredPayment.STOREDPAyMENTSList**

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the `ccrz__E_StoredPayment__c` record to create. You can specify the following fields:

account	endDate	startDate
accountNumber	expMonth	storefront
accountType	expYear	subaccountNumber
displayName	paymentType	token
enabled	sequence	user

 **Note:** By default, this method evaluates only one cart record at one time. Even though this key specifies a list, the method evaluates only the first item in the list.

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### **ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicStoredPaymentCreate.prepReturn` method removes this intermediate key from the working map.

SEE ALSO:

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

## ccrz.ccLogicStoredPaymentCreate.decorateStoredPayment

Evaluates the input fields and values and creates a corresponding `ccrz__E_StoredPayment__c` record, ready to insert.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> decorateStoredPayment (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicStoredPaymentCreate.processInput` method.

This method requires the following key in the working map:

#### **ccrz.ccApiStoredPayment.STOREDPAYMENTSLIST**

`List<Map<String, Object>>`, where the `Map<String, Object>` represents the stored payment fields and values to set on a `ccrz__E_StoredPayment__c` record.



**Note:** By default, this method evaluates only one cart record at one time. Even though this key specifies a list, the method evaluates only the first item in the list.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

#### **ccrz.ccApiStoredPayment.STOREDPAYMENTSLIST**

`List<ccrz__E_StoredPayment__c>` of the created stored payment record, ready to insert. If the input data didn't specify values for the following fields, this method sets default values:

Field	Source for Value
ccrz__Account__c	<a href="#">ccrz.cc_CallContext.currAccountId</a>
ccrz__EffectiveAccountID__c	<a href="#">ccrz.cc_CallContext.effAccountId</a>
ccrz__OwnerId__c	The current user's ID
ccrz__Storefront__c	<a href="#">ccrz.cc_CallContext.storefront</a>
ccrz__User__c	The current user's ID

SEE ALSO:

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

## ccrz.ccLogicStoredPaymentCreate.insertStoredPayment

Executes the DML `insert` operation for the created stored payment record, which assigns an ID to the record.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> insertStoredPayment (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the [ccrz.ccLogicStoredPaymentCreate.decorateStoredPayment](#) method.

This method requires the following key in the working map:

**ccrz.ccApiStoredPayment.STOREDPAyMENTSList**


`List<ccrz\_\_E\_StoredPayment\_\_c>` of the created stored payment record, ready to insert.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

**ccrz.ccApi.SUCCESS**

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

**ccrz.ccApiStoredPayment.ID**ID of the new `ccrz__E_StoredPayment__c` record.

SEE ALSO:

[Inserting and Updating Records](#)**ccrz.ccLogicStoredPaymentCreate.refetch**Calls the `ccrz.ccApiStoredPayment.fetch` method for fetching the stored payment record after it's created and inserted.**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global virtual Map<String, Object> refetch(Map<String, Object>)
```

**Inputs (Required)**

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicStoredPaymentCreate.insertStoredPayment` method.

This method requires the following key in the working map:



**ccrz.ccApi.SIZING**

Map<String, Object> that specifies [options for how the method returns data](#). For this method to execute, this key must include `ccrz.ccApi.SZ_REFETCH => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object> {
    ccrz.ccApiStoredPayment.ENTITYNAME => new Map<String, Object> {
        ccrz.ccApi.SZ_REFETCH => true
    }
}
```

 **Note:** This method respects any other data size keys specified in the original input's `ccrz.ccApi.SIZING` key.

## Outputs

This method returns the working Map<String, Object>, and sets the following key:

**ccrz.ccApiStoredPayment.STORED\_PAYMENT\_LIST**

List<Map<String, Object>>, where the Map<String, Object> represents the created `ccrz__E_StoredPayment__c` record, which the `ccrz.ccApiStoredPayment.fetch` method returns.

SEE ALSO:

[ccrz.ccApiStoredPayment.fetch](#)

## ccrz.ccLogicStoredPaymentCreate.prepReturn

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` from the `ccrz.ccLogicStoredPaymentCreate.refetch` method.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.

- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicStoredPaymentRemove

Define the business logic for deleting stored payment records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Global API Caller

`ccrz.ccApiStoredPayment.remove`

## Methods

IN THIS SECTION:

[process](#)

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

[processInput](#)

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

[fetchStoredPayment](#)

Calls the `ccrz.ccApiStoredPayment.fetch` method to return the stored payment records that you want to delete.

[deleteStoredPayment](#)

Executes the DML `delete` operation for the stored payment records.

[prepReturn](#)

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## ccrz.ccLogicStoredPaymentRemove.process

Overrides the base `ccrz.ccLogic.process` method, sets up an intermediate working map, and invokes the other methods for this class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

### Inputs (Required)

This method accepts a `Map<String, Object>`, which was the input data for the `ccrz.ccApiStoredPayment.remove` method.


### Logic Performed

This method invokes other methods of the `ccrz.ccLogicStoredPaymentRemove` class in the following order:

1. `processInput`
2. `fetchStoredPayment`
3. `deleteStoredPayment`
4. `prepReturn`

### Outputs

This method creates a `Map<String, Object>` that contains a copy of the original input data. This method passes this working map through the inner methods, and returns the final map after all inner methods have completed.

 **Note:** This method doesn't catch any exceptions that occur as the method executes.

## ccrz.ccLogicStoredPaymentRemove.processInput

Parses the working map for keys that downstream methods in the process chain require, and verifies that appropriate values are set.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> processInput (Map<String, Object>)
```

### Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccLogicStoredPaymentRemove.process` method.

This method checks that the input map specifies either of the following keys, and returns a `ccrz.ccApi.MissingInputException` if either key is missing:

**ccrz.ccApiStoredPayment.ID**

String that specifies the ID of a stored payment to remove.

**ccrz.ccApiStoredPayment.IDS**

`List<String>` that specifies the IDs of stored payments to remove.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following keys:

**ccrz.ccApi.API\_VERSION**

Integer that indicates which API version was used for the query.

**ccrz.ccApi.SUCCESS**

Boolean that's assigned an intermediate value of `false`. Subsequent methods in the logic chain update this value to `true` when the logic completes successfully.

**ccrz.ccLogic.ORIGINAL\_INPUT**

`Map<String, Object>` of the original, unmodified input data.

The `ccrz.ccLogicStoredPaymentRemove.prepReturn` method removes intermediate keys from the working map.

## ccrz.ccLogicStoredPaymentRemove.fetchStoredPayment

Calls the `ccrz.ccApiStoredPayment.fetch` method to return the stored payment records that you want to delete.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchStoredPayment (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicStoredPaymentRemove.processInput` method completes.

This method can accept either of the following parameters:

#### **ccrz.ccApiStoredPayment.ID**

String that specifies the ID of a stored payment to remove.

#### **ccrz.ccApiStoredPayment.IDS**

`List<String>` that specifies the IDs of stored payments to remove.

### Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

#### **ccrz.ccApiStoredPayment.STORED\_PAYMENT\_LIST**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_StoredPayment__c` record, which the `ccrz.ccApiStoredPayment.fetch` method returns.

#### **ccrz.ccApiSubscription.SUBSCRIPTIONS**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_Subscription__c` record associated with a returned `ccrz__E_StoredPayment__c` record.



**Note:** This output is returned only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12), and only when the original input data specifies `ccrz.ccApi.SZ_ASSOC => true`:

```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiStoredPayment.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_ASSOC => true
    }
}
```

```
}
}
```

SEE ALSO:

[ccrz.ccApiStoredPayment.fetch](#)

## ccrz.ccLogicStoredPaymentRemove.deleteStoredPayment

Executes the DML `delete` operation for the stored payment records.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> deleteStoredPayment (Map<String, Object>)
```

### Inputs (Required)

This method accepts the working `Map<String, Object>` after the [ccrz.ccLogicStoredPaymentRemove.fetchStoredPayment](#) method completes.

This method requires the following key in the working map:

#### **ccrz.ccApiStoredPayment.STOREDPAymentLIST**


`List<ccrz__E_StoredPayment__c>`, where each `Map<String, Object>` represents a `ccrz__E_StoredPayment__c` record to delete.

### Inputs (Optional)

If the working map includes the following key, this method evaluates it:

#### **ccrz.ccApiSubscription.SUBSCRIPTIONS**

`List<Map<String, Object>>`, where each `Map<String, Object>` represents a `ccrz__E_Subscription__c` record associated with a returned `ccrz__E_StoredPayment__c` record.

 **Note:** This input is evaluated only in B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12), and only when the original input data specifies `ccrz.ccApi.SZ_ASSOC => true`:


```
ccrz.ccApi.SIZING => new Map<String, Object>{
    ccrz.ccApiStoredPayment.ENTITYNAME => new Map<String, Object>{
        ccrz.ccApi.SZ_ASSOC => true
    }
}
```

## Outputs

This method returns the working `Map<String, Object>`, and sets the following key:

### `ccrz.ccApi.SUCCESS`

Boolean

Value	Usage
<code>true</code>	The call completed.
<code>false</code>	<p>The call encountered errors.</p> <p> <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code>, <a href="#">consider rolling back the API transaction to a previous savepoint</a>.</p>

SEE ALSO:

[Extension Examples for Stored Payments](#)

[Deleting Records](#)

## `ccrz.ccLogicStoredPaymentRemove.prepReturn`

Overrides the base `ccrz.ccLogic.prepReturn` method, removes intermediate keys added to the working map by the preceding methods, and returns the final return data keys.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> prepReturn (Map<String, Object>)
```

## Inputs (Required)

This method accepts the working `Map<String, Object>` after the `ccrz.ccLogicStoredPaymentRemove.deleteStoredPayment` method completes.

## Outputs

This method removes the following intermediate key from the working map, and returns the final `Map<String, Object>`.


- `ccrz.ccLogic.ORIGINAL_INPUT`

## ccrz.ccLogicUserCreateUser

Define the business logic for creating a user record from the fields and values that you specify. This class doesn't insert the user. Instead, the `ccrz.ccLogicUserRegister` logic service provider class invokes the `Site.createExternalUser` method.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicUserCreateUser](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiUser.createUser](#)


[ccrz.ccLogicUserRegister](#)

## ccrz.ccLogicUserGetProfileData

Define the business logic for returning user, contact, and account record data for populating fields on a buyer's My Account page.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicUserGetProfileData](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiUser.getProfileData](#)


## ccrz.ccLogicUserLogin

Define the business logic for authenticating a user into a storefront.

This content is available only as an archived PDF from CloudCraze documentation:



(PDF) [ccrz.ccLogicUserLogin](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:


[ccrz.ccApiUser.login](#)

## ccrz.ccLogicUserRegister

Define the business logic for creating the user, contact, and account records for a buyer who's registering on your storefront.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicUserRegister](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:


[ccrz.ccApiUser.registerUser](#)

## ccrz.ccLogicUserUpdateProfileData

Define the business logic for updating user or contact records with new values that a buyer specifies on the My Account page.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicUserUpdateProfileData](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiUser.updateProfileData](#)

## ccrz.ccLogicWLAddTo

Define the business logic for adding products to a wish list.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicWLAddTo](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiWishList](#)

*Salesforce Help* [Allow a Buyer to Create a Wish List of Items for a Future Cart](#)

## ccrz.ccLogicWLCreate

Define the business logic for creating a wish list.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicWLCreate](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiWishList](#)

*Salesforce Help* [Allow a Buyer to Create a Wish List of Items for a Future Cart](#)

## ccrz.ccLogicWLCreateCart

Define the business logic for creating a cart record from an existing wish list.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicWLCreateCart](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiWishList](#)

*Salesforce Help* [Allow a Buyer to Create a Wish List of Items for a Future Cart](#)

## ccrz.ccLogicWLGetActive

Define the business logic for resolving which wish list is active for the current user, account (including effective accounts), storefront, and currency.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicWLGetActive](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiWishList](#)

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)

## ccrz.ccLogicWLRemove

Define the business logic for deleting a wish list.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicWLRemove](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiWishList](#)

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)

## ccrz.ccLogicWLRemoveFrom

Define the business logic for deleting wish list items from a specified wish list.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicWLRemoveFrom](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiWishList](#)


[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)

## ccrz.ccLogicWLSetActive

Define the business logic for setting a single wish list as the active wish list for a particular owner, and updating all the owner's other wish lists to inactive.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccLogicWLSetActive](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.ccApiWishList](#)

[Salesforce Help Allow a Buyer to Create a Wish List of Items for a Future Cart](#)


## REST API Endpoints for B2B Commerce for Visualforce

---

B2B Commerce for Visualforce provides REST API endpoints for most of its global Apex API methods.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [B2B Commerce for Visualforce REST APIs](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.


## Create Subscriber APIs for B2B Commerce for Visualforce

---

As part of your organization's customization and extension strategy, you can develop custom APIs that build on the default Apex classes and methods that B2B Commerce for Visualforce provides.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [Creating Subscriber APIs](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## Global Extension Point Classes for B2B Commerce for Visualforce

---

B2B Commerce for Visualforce provides several *global extension points*: Apex classes that are called from other code in the managed package and execute specific business logic for certain storefront functionality. For example, several extension point classes define behavior for loading data during checkout.

 **Note:**

- Some extension point classes are deprecated, but are still available in the managed package for compatibility with legacy subscriber code.
- Sometimes, B2B Commerce for Visualforce updates extension point classes in a new version of the B2B Commerce managed package. In subscriber code, you reference the latest version of these classes that's available in your version of the managed package.

## IN THIS SECTION:

[Extend a Global Extension Point Class](#)

To implement custom business logic for your organization, you can write subscriber code that extends a default extension point class.

[ccrz.cc\\_api\\_CartExtension](#)

Implement subscriber-defined behavior for processing and validating cart data. This class defines a mixture of legacy and current behavior, and the `ccrz.ccLogicCartValidate` logic service provider class replaces the legacy functionality from this class.

[ccrz.cc\\_api\\_DeliveryDate](#)

Implement subscriber-defined behavior for showing an account's available delivery dates during checkout. On the storefront, the Checkout page calls this extension point class before rendering shipping options.

[ccrz.cc\\_api\\_OutboundOrderCancel](#)

Implement subscriber-defined behavior for determining whether a buyer can cancel an order from the My Orders section of the My Account page.

[ccrz.cc\\_api\\_PriceAdjustment](#)

Implement subscriber-defined behavior for customizing price information on legacy versions of the Shopping Cart page or Checkout page.

[ccrz.cc\\_api\\_ProductQuantityRule](#)

Define the quantity increments that a buyer can only use when adding or updating cart items. For example, maybe you want to require buyers to purchase a certain product only in increments of five.

[ccrz.cc\\_api\\_ShippingAndHandling](#)

Implement subscriber-defined behavior for populating the available shipping methods during checkout. On the storefront, the Checkout page calls this extension point class before rendering shipping options.

[ccrz.cc\\_hk\\_Catalog](#)

Return info from your product catalog used for populating autocomplete suggestions on the storefront.

[ccrz.cc\\_hk\\_Category \(Deprecated\)](#)

Dynamically construct the category tree for a storefront by querying cached category data.

[ccrz.cc\\_hk\\_DynamicTheme](#)

Load a unique storefront theme based on the current buyer's account group.

[ccrz.cc\\_hk\\_EffectiveAccount](#)

Return and filter effective accounts that a buyer can access other than the buyer's default assigned account. Update the active shopping cart for a selected effective account.

[ccrz.cc\\_hk\\_Invoice](#)

Create and return data for an invoice after a buyer completes a storefront purchase, and process payments associated with invoices.

[ccrz.cc\\_hk\\_Menu](#)

Dynamically construct the menu hierarchy for a storefront's navigation bar by querying cached menu data.

[ccrz.cc\\_hk\\_Order](#)

Implement subscriber-defined behavior for placing an order, reordering, and showing order history. This class defines a mixture of legacy and current behavior, and the order logic service provider classes replace the legacy functionality from this class.

[ccrz.cc\\_hk\\_Payment](#)

Evaluate and maintain stored payments and transaction payments.

[ccrz.cc\\_hk\\_Pricing](#)

Implement subscriber-defined behavior for pricing products, including entitlements, contracts, and other price rules. This class defines a legacy behavior. For new implements, extend the `ccrz.ccLogicProductPricing` logic service provider instead.

[ccrz.cc\\_hk\\_SSO](#)

Implement subscriber-defined behavior for overriding default storefront URLs, such as to add a single sign-on (SSO) service for logging in.

[ccrz.cc\\_hk\\_Subscriptions](#)

Implement subscriber-defined behavior for overriding the default business logic for processing subscriptions and creating installment orders.

[ccrz.cc\\_hk\\_TaxCalculation](#)

Calculate the tax amount on a particular cart from the Order Review section of the storefront's Checkout page. The calculated tax is based on the taxable products included in the cart and the cart's shipping address.

[ccrz.cc\\_hk\\_UserInterface](#)

Specify the JavaScript and CSS libraries referenced in each storefront page, including standard includes, proprietary libraries, and theme-specific files in a static resource.

## Extend a Global Extension Point Class

To implement custom business logic for your organization, you can write subscriber code that extends a default extension point class.

For example, let's say that you want to extend the default behavior for returning a user's available effective accounts.

1. In your org, create an Apex class for your customization.

```
global with sharing class cc_example_hk_EffectiveAccount extends
ccrz.cc_hk_EffectiveAccount { }
```



### Tip:

- We recommend that you define and adopt a consistent naming convention for your organization's subscriber code. Include a common prefix, your project's name, and the name of the default class, such as `cc_projectName_defaultClassName`.
  - Specify an appropriate [sharing keyword](#), depending on the availability and permissions that you want the class to have.
  - Use the `extends` keyword since you're extending a default extension point class.
2. Inside the class definition, override the specific method of the `ccrz.cc_hk_EffectiveAccount` extension point class whose behavior you want to customize.

Let's say that you want to override the `fetchAccounts` method, which returns the list of accounts that a buyer can access.

```
global with sharing class cc_example_hk_EffectiveAccount extends
ccrz.cc_hk_EffectiveAccount {
    global override Map<String, Object> fetchAccounts (Map<String, Object> inputData) {
    }
}
```

Because each extension point method accepts a `Map<String, Object>`, your method override must also accept a `Map<String, Object>` for input data.

3. Inside your method override, add your custom business logic. Make sure that you include a return statement at the end of your method override.

```
global with sharing class cc_example_hk_EffectiveAccount extends
ccrz.cc_hk_EffectiveAccount {
    global virtual override Map<String, Object> fetchAccounts(Map<String, Object>
) {
        // Your custom logic
        return inputData;
    }
}
```

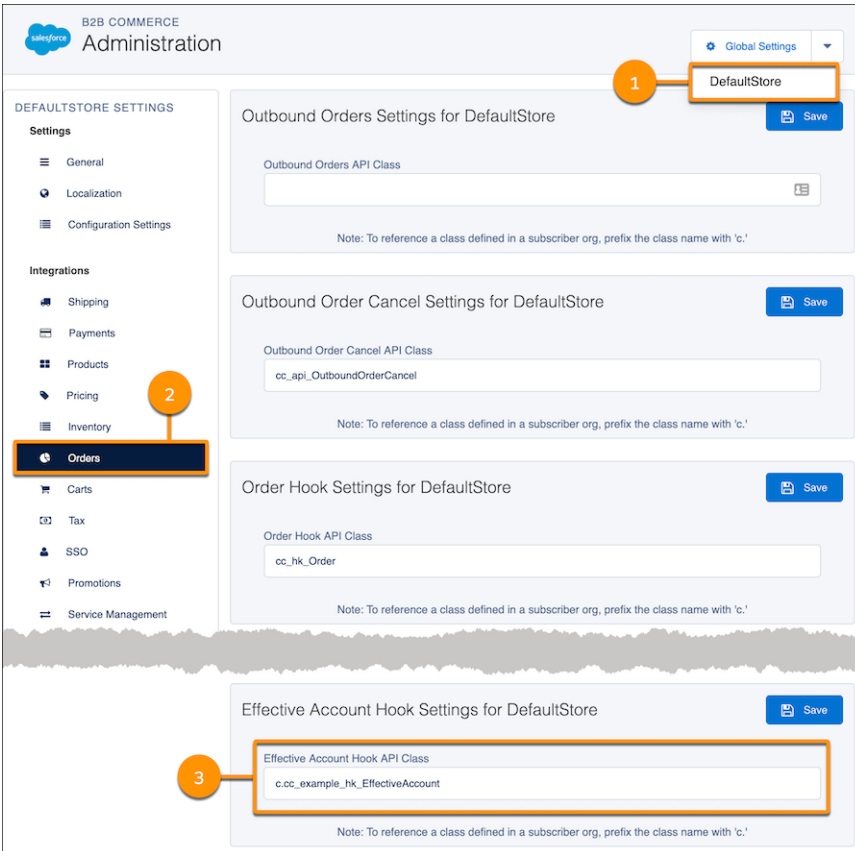
 Tip:


- Since you're overriding a `virtual` method, you can [invoke the `super` keyword in your custom logic](#).
- To help you understand what data a method expects and returns by default, check out the reference for that particular method. During development, you can also serialize the default input and output data within your method override. This step can help you make sure that your extension code handles all expected inputs and outputs correctly so that your code executes successfully.

```
global with sharing class cc_example_hk_EffectiveAccount extends
ccrz.cc_hk_EffectiveAccount {
    global virtual override Map<String, Object> fetchAccounts(Map<String, Object>
) {
        System.debug('-- inputData ' + JSON.serialize(inputData));
        Map<String, Object> returnData = super.fetchAccounts(inputData);
        System.debug('-- returnData ' + JSON.serialize(returnData));

        return returnData;    }
}
```

4. Save your class, and resolve any compilation errors.
5. Update your storefront settings so that B2B Commerce for Visualforce references your override instead of the default extension point class.



- a. On the CC Admin tab in your org, select your storefront (1).
- b. In the Storefront Settings menu, scroll the section under Integrations that corresponds to the extension point that you're customizing (2). For example, for the `ccrz.cc_hk_EffectiveAccount` extension point class, select **Orders** from the Storefront Settings menu.
  -  **Tip:** For help with locating which section of your storefront settings contains the extension point you're modifying, see the reference for that extension point class.
- c. Find the settings for the extension point you're modifying, such as Effective Account Hook API Class (3).
- d. Enter the name of your custom class, prefixed with `c.`, such as `c.cc_example_hk_EffectiveAccount`, and click **Save**.

## ccrz.cc\_api\_CartExtension

Implement subscriber-defined behavior for processing and validating cart data. This class defines a mixture of legacy and current behavior, and the `ccrz.ccLogicCartValidate` logic service provider class replaces the legacy functionality from this class.

## Compatibility

This reference applies to:




Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

In these versions of the managed package, the `ccrz.ccLogicCartValidate` class provides methods that replace some methods of this extension point class:

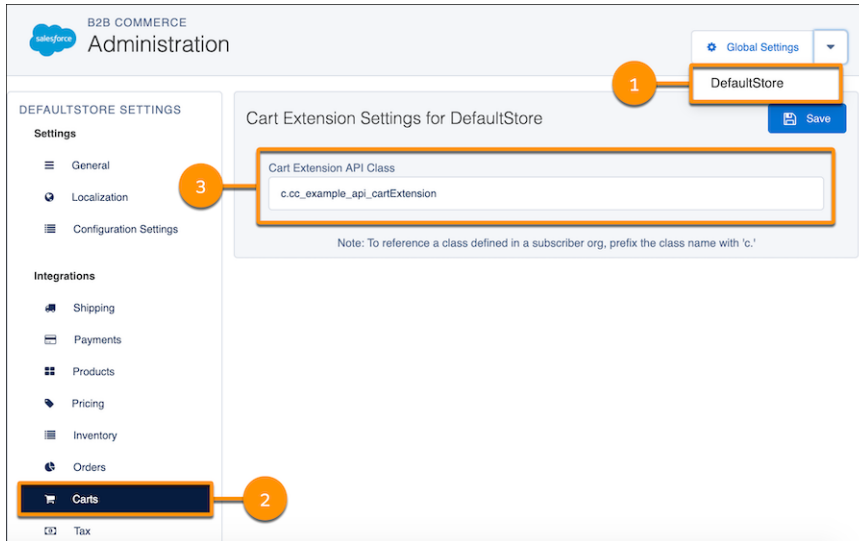
Legacy Method	Replacement Method
<code>ccrz.cc_api_CartExtension.allowCheckout</code>	<code>ccrz.ccLogicCartValidate.processValidate</code>
<code>ccrz.cc_api_CartExtension.getCartMessages</code>	<code>ccrz.ccLogicCartValidate.processValidate</code>
<code>ccrz.cc_api_CartExtension.preprocess</code>	None. This functionality remains current in the managed package.
<code>ccrz.cc_api_CartExtension.prepareForSave</code>	None. This functionality remains current in the managed package.
<code>getInstance</code>	None. This functionality remains current in the managed package.

To continue referencing the legacy `ccrz.cc_api_CartExtension.allowCheckout` or `ccrz.cc_api_CartExtension.getCartMessages` methods when loading cart data on your storefront's Shopping Cart page, override the default `ccrz.ccLogicCartValidate.processValidate` method. In the subscriber code for your override, call these legacy methods. The managed package provides an [example extension](#), `ccrz.ccLogicCartValidate7Hk`, that you can use as a starting point for overriding `ccrz.ccLogicCartValidate.processValidate`.

 **Important:** In general, we recommend that new solutions extend methods of `ccrz.ccLogicCartValidate` instead of extending methods of the `ccrz.cc_api_CartExtension` class.

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Carts** in the Integrations section of the Storefront Settings menu (2). Update the Cart Extension API Class field (3) with your custom class name, prefixed with `c.`, such as `c.cc_example_api_CartExtension`.



 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

## Methods

### IN THIS SECTION:

#### [allowCheckout](#)

Determines whether the current user can check out from the Shopping Cart page.

#### [getCartMessages](#)

Provides an extension point for showing errors, warnings, or informational messages on the Shopping Cart page.

#### [preprocess](#)

Provides an extension point for evaluating any rules, requirements, or considerations that you want to enforce before the Shopping Cart page loads or updates cart data. For example, maybe you want to automatically add or remove items from the cart based on the cart's current contents.

#### [prepareForSave](#)

Provides an extension point for saving and processing custom data when the buyer clicks Update or Checkout on the Shopping Cart page. For example, override this method to save data from custom fields or objects, apply surcharges, or trigger other actions or services.

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_CartExtension` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## `ccrz.cc_api_CartExtension.allowCheckout`

Determines whether the current user can check out from the Shopping Cart page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Important:

- This method defines legacy behavior, and by default the managed package instead references the [ccrz.ccLogicCartValidate.processValidate method](#). To reference this method in your solution, override the default `ccrz.ccLogicCartValidate.processValidate` method so that it calls this method. The managed package provides an [example extension](#), `ccrz.ccLogicCartValidate7Hk`, that you can use as a starting point for overriding `ccrz.ccLogicCartValidate.processValidate`.
- When you use `ccrz.ccLogicCartValidate7Hk` or your own extension, the Shopping Cart page calls this method when the page first loads. The page also calls this method when a buyer clicks Update or Checkout.

## Signature

```
global virtual Boolean allowCheckout (Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### **ccrz.cc\_api\_CartExtension.PARAM\_CARTBEAN**

`ccrz.cc_bean_CartSummary` that represents the current `ccrz__E_Cart__c` record.

### **ccrz.cc\_api\_CartExtension.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### **ccrz.cc\_api\_CartExtension.PARAM\_USERID**

ID of the user assigned to the cart:

- When an authenticated buyer is logged in, this value is the ID of their user record.
- When a customer service representative (CSR) is logged in on behalf of a buyer, this value is the buyer's user ID.

## Outputs

Boolean

Value	Usage
true	The current user is allowed to check out.
false	The current user isn't allowed to check out.

## SEE ALSO:

[ccrz.ccLogicCartValidate.processValidate](#)

[ccrz.ccLogicCartValidate7Hk Example Extension Class](#)

[ccrz.cc\\_bean\\_CartSummary](#)

## ccrz.cc\_api\_CartExtension.getCartMessages

Provides an extension point for showing errors, warnings, or informational messages on the Shopping Cart page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Important:

- This method defines legacy behavior, and by default the managed package instead references the [ccrz.ccLogicCartValidate.processValidate method](#). To reference this method in your solution, override the default `ccrz.ccLogicCartValidate.processValidate` method so that it calls this method. The managed package provides an [example extension](#), `ccrz.ccLogicCartValidate7Hk`, that you can use as a starting point for overriding `ccrz.ccLogicCartValidate.processValidate`.
- When you use `ccrz.ccLogicCartValidate7Hk` or your own extension, the Shopping Cart page calls this method when the page first loads. The page also calls this method when a buyer clicks Update or Checkout.

### Signature

```
global virtual List<ccrz.cc_bean_Message> getCartMessages (Map<String, Object>)
```

### Inputs (Optional)

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_api\_CartExtension.PARAM\_CARTBEAN**

[ccrz.cc\\_bean\\_CartSummary](#) that represents the current [ccrz\\_\\_E\\_Cart\\_\\_c](#) record.

**ccrz.cc\_api\_CartExtension.PARAM\_STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to [ccrz.cc\\_CallContext.storefront](#).

**ccrz.cc\_api\_CartExtension.PARAM\_USERID**

ID of the user assigned to the cart:

- When an authenticated buyer is logged in, this value is the ID of their user record.
- When a customer service representative (CSR) is logged in on behalf of a buyer, this value is the buyer's user ID.

## Outputs

List<[ccrz.cc\\_bean\\_Message](#)> of messages to show to the buyer.

SEE ALSO:

[ccrz.ccLogicCartValidate.processValidate](#)

[ccrz.ccLogicCartValidate7Hk Example Extension Class](#)

[ccrz.cc\\_bean\\_CartSummary](#)

[ccrz.cc\\_bean\\_Message](#)


## ccrz.cc\_api\_CartExtension.preprocess

Provides an extension point for evaluating any rules, requirements, or considerations that you want to enforce before the Shopping Cart page loads or updates cart data. For example, maybe you want to automatically add or remove items from the cart based on the cart's current contents.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method defines current behavior, and isn't replaced by any [ccrz.ccLogicCartValidate](#) method. When you use the [ccrz.ccLogicCartValidate7Hk](#) example extension or your own extension, the Shopping Cart page still calls this method when the page first loads. The page also calls this method when a buyer clicks Update or Checkout.

## Signature

```
global virtual Map<String, Object> preprocess (Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_api_CartExtension.PARAM_CART`

`ccrz__E_Cart__c` that represents the cart to load or update.

### `ccrz.cc_api_CartExtension.PARAM_CARTITEMS`

List<`ccrz__E_CartItem__c` that represents the items to load or update in the cart.

### `ccrz.cc_api_CartExtension.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### `ccrz.cc_api_CartExtension.PARAM_USERID`

ID of the user assigned to the cart:

- When an authenticated buyer is logged in, this value is the ID of their user record.
- When a customer service representative (CSR) is logged in on behalf of a buyer, this value is the buyer's user ID.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_api_CartExtension.PARAM_CART`

`ccrz__E_Cart__c` that represents the loaded or updated cart.

### `ccrz.cc_api_CartExtension.PARAM_CARTITEMS`

List<`ccrz__E_CartItem__c` that represents the loaded or updated cart items. This list is the same list that you specified in the input map, but updated with any changes that the method completed. For example:

- If you increment the quantity for some of the passed-in cart items, return the updated cart items in this key.
- If you add cart items within this method, we recommend that you include the added items in this key. However, the method doesn't strictly require that you do so.



**Tip:**

- To modify the cart or cart items within this method, use [Data Manipulation Language \(DML\)](#) operations.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz.ccApiCart.price](#)

[ccrz.ccLogicCartValidate7Hk Example Extension Class](#)


## ccrz.cc\_api\_CartExtension.prepareForSave

Provides an extension point for saving and processing custom data when the buyer clicks Update or Checkout on the Shopping Cart page. For example, override this method to save data from custom fields or objects, apply surcharges, or trigger other actions or services.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method defines current behavior. However, for new implementations, we recommend that you override the `ccrz.ccLogicCartSave.updateBeforeSave` method instead.

### Signature

```
global virtual Map<String, Object> prepareForSave (Map<String, Object>)
```

### Inputs (Optional)

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_api\_CartExtension.PARAM\_CART**

`ccrz__E_Cart__c` that represents the cart to update.

**ccrz.cc\_api\_CartExtension.PARAM\_CARTBEAN**

`ccrz.cc_bean_CartSummary` that represents the current `ccrz__E_Cart__c` record.

**ccrz.cc\_api\_CartExtension.PARAM\_CARTITEMS**

`List<ccrz__E_CartItem__c` that represents the items to update in the cart.

**ccrz.cc\_api\_CartExtension.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.cc\_api\_CartExtension.PARAM\_USERID**

ID of the user assigned to the cart:

- When an authenticated buyer is logged in, this value is the ID of their user record.
- When a customer service representative (CSR) is logged in on behalf of a buyer, this value is the buyer's user ID.

## Outputs

None

SEE ALSO:

[ccrz.ccLogicCartSave.updateBeforeSave](#)

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

## ccrz.cc\_api\_CartExtension.getInstance

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_CartExtension` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static ccrz.cc_api_cartExtension getInstance(Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### **ccrz.cc\_api\_CartExtension.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_api_CartExtension`, or your custom extension point class.



## ccrz.cc\_api\_DeliveryDate

Implement subscriber-defined behavior for showing an account's available delivery dates during checkout. On the storefront, the Checkout page calls this extension point class before rendering shipping options.

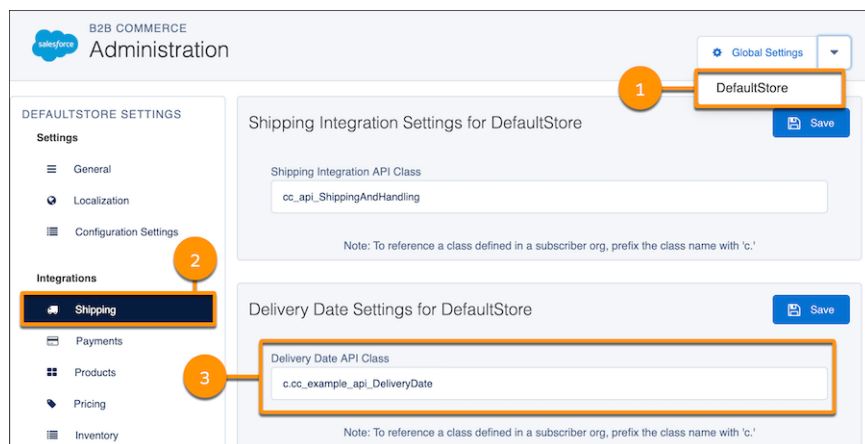
### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Extension Class Setting in CC Admin


If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Shipping** in the Integrations section of the Storefront Settings menu (2). Update the Delivery Date API Class field (3) with your custom class name, prefixed with *c.*, such as *c.cc\_example\_api\_DeliveryDate*.



**Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

### Front-End Input

On the storefront, a buyer selects dates from a date picker widget, which by default uses the `bootstrap-datepicker.js` library, referenced from the `ccrz.cc_hk_UserInterface` extension point class. To define rules for which dates are available for a buyer to select from the widget, use the `getDeliveryDates` method of this class.

 **Note:** This library replaced the `zebra_datepicker.js` in B2B Commerce for Visualforce Summer '18 (version 4.9). If you've migrated from a version that's older than Summer '18, update the `Date_Format` page label to use the [bootstrap-datepicker.js standard date format](#).

## Methods

### IN THIS SECTION:

#### [getDeliveryDates](#)

Returns available delivery dates for the date picker widget on the storefront. To specify your solution's rules for available delivery dates, override this method in your subscriber code.

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_DeliveryDate` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

### SEE ALSO:

[ccrz.cc\\_hk\\_UserInterface](#)

[Salesforce Help Set Up the Requested Delivery Date Picker on the Checkout Page](#)

[Salesforce Help Default JavaScript Libraries for Storefront Pages](#)

[bootstrap-datepicker Documentation](#)

[Zebra Datepicker Documentation](#)

## ccrz.cc\_api\_DeliveryDate.getDeliveryDates

Returns available delivery dates for the date picker widget on the storefront. To specify your solution's rules for available delivery dates, override this method in your subscriber code.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> getDeliveryDates(ID, String,
ccrz.cc_bean_MockContactAddress)
```

## Inputs (Required)

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_api\_DeliveryDate.ACCOUNT\_ID**

ID of the account record used for checkout.

**ccrz.cc\_api\_DeliveryDate.CART\_ID**


String that specifies the ID of the cart record used for checkout.

**ccrz.cc\_api\_DeliveryDate.SHIP\_ADDRESS**

`ccrz.cc_bean_MockContactAddress` that represents the shipping address selected for checkout.

## Outputs

`Map<String, Object>` that can include the following keys:

 **Note:** In B2B Commerce for Visualforce Summer '18 (version 4.9), the `bootstrap-datepicker.js` library replaced the `zebra_datepicker.js` library for rendering the date picker widget on the storefront. The return data keys of this method still specify delivery date options that correspond to `zebra_datepicker.js` configuration options. The `cc_CheckoutShippingRD.component` Visualforce component in the managed package translates the data that this method returns into a format that `bootstrap-datepicker.js` can use.

**ccrz.cc\_api\_DeliveryDate.API\_ERROR**

Boolean

Value	Usage
<code>true</code>	An error occurred when constructing the delivery date data. When this key is <code>true</code> and the Validate Delivery Date setting of the Shipping Options storefront configuration is also <code>true</code> , the checkout flow doesn't continue.
<code>false</code> (default)	The method returned the delivery date data successfully.

**ccrz.cc\_api\_DeliveryDate.DURATION**

Integer that specifies the number of days, after the offset value, when dates are no longer selectable in the date picker widget. For example, a duration value of 365 specifies that the final selectable date is one year after the first selectable date.

**ccrz.cc\_api\_DeliveryDate.EXCLUDED\_DATES**

`List<String>` of values for the `disabled_dates` parameter of the `zebra_datepicker.js` library. This list specifies dates that you want to disable in the date picker widget. Each date uses the format **dd MM yyyy**, such as 31 12 2020.

 **Note:**

- The `cc_CheckoutShippingRD.component` Visualforce component converts these values to the **mm/dd/yyyy** format, such as 12/31/2020, and sets the corresponding `datesDisabled` option of the `bootstrap-datepicker.js` library.
- You can provide either `ccrz.cc_api_DeliveryDate.EXCLUDED_DATES` or `ccrz.cc_api_DeliveryDate.INCLUDED_DATES`, but not both.

**ccrz.cc\_api\_DeliveryDate.INCLUDED\_DATES**

`List<String>` of values for the `enabled_dates` parameter of the `zebra_datepicker.js` library. This list specifies dates that you want to enable in the date picker widget. Each date uses the format **dd MM yyyy**, such as 31 12 2020.

 **Note:**

- The `cc_CheckoutShippingRD.component` Visualforce component converts these values to the `mm/dd/yyyy` format, such as 12/31/2020. Because the `bootstrap-datepicker.js` library doesn't provide a corresponding option, `cc_CheckoutShippingRD.component` includes a workaround to set these values.
- You can provide either `ccrz.cc_api_DeliveryDate.EXCLUDED_DATES` or `ccrz.cc_api_DeliveryDate.INCLUDED_DATES`, but not both.

**ccrz.cc\_api\_DeliveryDate.MESSAGES**

List<[ccrz.cc\\_bean\\_Message](#)> of messages to show to the buyer.

**ccrz.cc\_api\_DeliveryDate.OFFSET**

Integer that specifies the number of days from today until the first selectable date in the date picker widget. For example, an offset value of 1 specifies that the first selectable date is tomorrow.


The managed package also defines the following keys, but doesn't use them for returning any data:

- `ccrz.cc_api_DeliveryDate.DAYS_FILTER`
- `ccrz.cc_api_DeliveryDate.MAX_DATE`
- `ccrz.cc_api_DeliveryDate.MIN_DATE`

 **Example: Examples**


Allow a buyer to select only Monday, Wednesday, or Friday for a delivery date. The range begins tomorrow, and is available for one year.

```
global class cc_example_api_DeliveryDateMWF extends ccrz.cc_api_DeliveryDate{
    global override Map<String, Object> getDeliveryDates (Map<String, Object>
deliveryDatesInputData) {
        return new Map<String, Object> {
            ccrz.cc_api_DeliveryDate.OFFSET => 1,
            ccrz.cc_api_DeliveryDate.DURATION => 365,
            ccrz.cc_api_DeliveryDate.INCLUDED_DATES => new List<String>{'* * * 1,3,5'}
        };
    }
}
```

 **Tip:** The value `'* * * 1,3,5'` uses syntax for the `enabled_dates` parameter of the `zebra_datepicker.js` library. For more examples and other supported syntax, see the `zebra_datepicker.js` documentation.

Allow a buyer to select any weekday for a delivery date. The range begins the day after tomorrow, and is available for 90 days.

```
global class cc_example_api_DeliveryDateWeekdays extends ccrz.cc_apiDeliveryDate{
    global override Map<String, Object> getDeliveryDates (Map<String, Object>
deliveryDatesInputData) {
        return new Map<String, Object> {
            ccrz.cc_api_DeliveryDate.OFFSET => 2,
            ccrz.cc_api_DeliveryDate.DURATION => 90,
            ccrz.cc_api_DeliveryDate.EXCLUDED_DATES => new List<String>{'* * * 0,6'}
        };
    }
}
```

 **Tip:** The value `'* * * 0,6'` uses syntax for the `disabled_dates` parameter of the `zebra_datepicker.js` library. The `cc_CheckoutShippingRD.component` Visualforce component converts these values for the corresponding `datesDisabled` option of the `bootstrap-datepicker.js` library.

#### SEE ALSO:

[ccrz.ccApiCart.getDeliveryDates](#)

[ccrz.ccLogicCartGetDeliveryDates](#)

[Salesforce Help](#) Set Up the Requested Delivery Date Picker on the Checkout Page

[Salesforce Help](#) List of Configuration Modules and Settings

[Salesforce Help](#) Default JavaScript Libraries for Storefront Pages

[bootstrap-datepicker](#) Documentation

[Zebra Datepicker](#) Documentation

## ccrz.cc\_api\_DeliveryDate.getInstance

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_DeliveryDate` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global static ccrz.cc_api_DeliveryDate getInstance (Map<String, Object>)
```

### Inputs (Optional)

`Map<String, Object>` that can include the following keys:

#### **ccrz.cc\_api\_DeliveryDate.STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_api_DeliveryDate`, or your custom extension point class.

## ccrz.cc\_api\_OutboundOrderCancel

Implement subscriber-defined behavior for determining whether a buyer can cancel an order from the My Orders section of the My Account page.

## Compatibility


This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Orders** in the Integrations section of the Storefront Settings menu (2). Update the Outbound Order Cancel API Class field (3) with your custom class name, prefixed with `c.`, such as `c.cc_example_api_OutboundOrderCancel`.

The screenshot shows the B2B Commerce Administration interface. The top navigation bar includes the Salesforce logo and 'B2B COMMERCE Administration'. A 'Global Settings' dropdown menu (1) is open, showing 'DefaultStore'. The left sidebar has 'Settings' and 'Integrations' sections. Under 'Integrations', the 'Orders' option (2) is selected. The main content area shows 'Outbound Orders Settings for DefaultStore' and 'Outbound Order Cancel Settings for DefaultStore'. The 'Outbound Order Cancel API Class' field (3) in the latter section contains the text 'c.cc\_example\_api\_OutboundOrderCancel'. A 'Save' button is visible next to the field.

 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

## Front-End Input

On the storefront, a buyer clicks Cancel on an order from the My Orders section of the My Account page. For help with configuring the shipping options on the Checkout page, see [https://help.salesforce.com/articleView?id=b2b\\_commerce\\_page\\_my\\_account\\_my\\_orders.htm](https://help.salesforce.com/articleView?id=b2b_commerce_page_my_account_my_orders.htm).

## Methods



**Note:** In addition to the following methods, this class defines another method, `cancelOrder`, that is deprecated. Instead, use the `ccrz.ccApiOrder.cancel` global API method.

### IN THIS SECTION:

#### [eligibleForCancellation](#)

Determines whether a buyer can cancel an order.

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_OutboundOrderCancel` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

### SEE ALSO:

[ccrz.ccApiOrder.canCancel](#)

[ccrz.ccApiOrder.cancel](#)

[Salesforce Help Allow a Buyer to Manage Their Orders on the My Account Page](#)

## ccrz.cc\_api\_OutboundOrderCancel.eligibleForCancellation

Determines whether a buyer can cancel an order.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Boolean eligibleForCancellation (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that can include the following keys:

**`ccrz.cc_api_OutboundOrderCancel.ORDER_DATA_PARAM`**

`ccrz.cc_bean_MockOrder` that represents the `ccrz__E_Order__c` record to evaluate.

**`ccrz.cc_api_OutboundOrderCancel.ORDER_OBJ_PARAM`**

`ccrz__E_Order__c` to evaluate.

## Outputs

Boolean

Value	Usage
<code>true</code>	The buyer can cancel the order. By default, this method returns <code>true</code> when the order's <code>ccrz__OrderStatus__c</code> field specifies one of the following values: <ul style="list-style-type: none"> <li>Cancelled</li> <li>Pending Cancel</li> <li>Shipped</li> </ul>
<code>false</code>	The buyer can't cancel the order.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz.cc\\_bean\\_MockOrder](#)

## `ccrz.cc_api_OutboundOrderCancel.getInstance`

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_OutboundOrderCancel` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



## Signature

```
global static ccrz.cc_api_OutboundOrderCancel getInstance(Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### **ccrz.cc\_api\_OutboundOrderCancel.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_api_OutboundOrderCancel`, or your custom extension point class.

## ccrz.cc\_api\_PriceAdjustment

Implement subscriber-defined behavior for customizing price information on legacy versions of the Shopping Cart page or Checkout page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

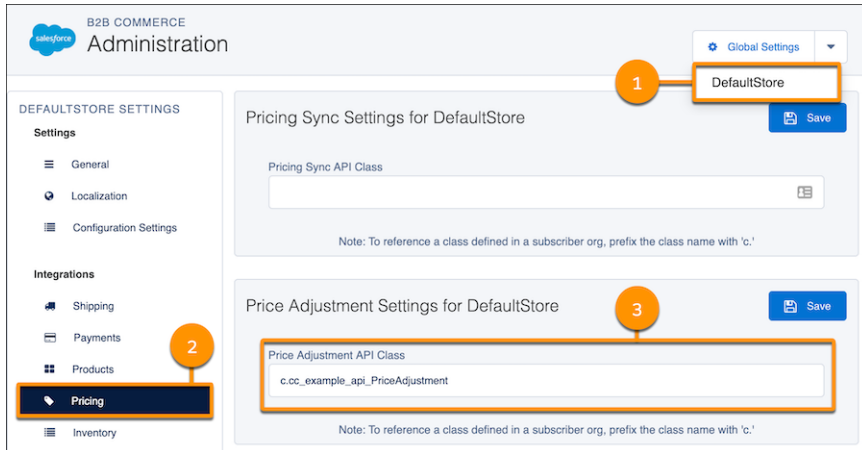
This class is compatible only with the following storefront page versions:

- Shopping Cart page version 1.0. Version 2.0 was introduced in [B2B Commerce for Visualforce Spring '19 \(version 4.10\)](#), but version 1.0 remains available for compatibility.
- Version 1.0 of the Order Review section of the Checkout page. Version 2.0 was introduced in [B2B Commerce for Visualforce Summer '19 \(version 4.11\)](#), but version 1.0 remains available for compatibility.

When you use version 2.0 of the Shopping Cart page or Order Review section of the Checkout page, extend the `ccrz.ccLogicCartPrice` logic service provider class instead.

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Pricing** in the Integrations section of the Storefront Settings menu (2). Update the Price Adjustment API Class field (3) with your custom class name, prefixed with *c.*, such as *c.cc\_example\_api\_PriceAdjustment*.



 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

## Front-End Input

On the storefront, the methods of this class are called:

- When the Shopping Cart page version 1.0 first loads, and whenever the cart is updated.
- Before version 1.0 of the Order Review section of the Checkout page renders.

## Methods

IN THIS SECTION:

### [computePricingCart](#)

Evaluates a `ccrz.cc_bean_CartSummary` input object when the Shopping Cart page loads, and sets the bean's price information. This method executes after the standard price logic.

### [computePricingReview](#)

Evaluates a cart when the Order Review section of the Checkout page loads, and returns any applicable error messages to show to the buyer.

[getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_PriceAdjustment` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

SEE ALSO:

[ccrz.ccLogicCartPrice](#)

[Salesforce Help Show Fees and Charges in the Order Review Section of the Checkout Page](#)


## ccrz.cc\_api\_PriceAdjustment.computePricingCart

Evaluates a `ccrz.cc_bean_CartSummary` input object when the Shopping Cart page loads, and sets the bean's price information. This method executes after the standard price logic.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Important:** This method defines legacy behavior, and by default the managed package instead references the methods of the `ccrz.ccLogicCartPrice` class.

### Signature

```
global virtual Map<String, Object> computePricingCart (Map<String, Object>)
```

### Inputs (Optional)

`Map<String, Object>` that can include the following keys:


**ccrz.cc\_api\_PriceAdjustment.PARAM\_CARTBEAN**

`ccrz.cc_bean_CartSummary` that represents the current `ccrz__E_Cart__c` record.

 **Warning:** Change data on the bean instead of executing DML operations to the underlying `ccrz__E_Cart__c` data.

**ccrz.cc\_api\_PriceAdjustment.PARAM\_CURRENCYCODE**

String that specifies an ISO 4217 currency code, such as `USD` or `JPY`.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userCurrency`.

**ccrz.cc\_api\_PriceAdjustment.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**ccrz.cc\_api\_PriceAdjustment.PARAM\_USERID**

ID of the user assigned to the cart:

- When an authenticated buyer is logged in, this value is the ID of their user record.
- When a customer service representative (CSR) is logged in on behalf of a buyer, this value is the buyer's user ID.

## Outputs

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_api\_PriceAdjustment.PARAM\_CARTBEAN**

`ccrz.cc_bean_CartSummary` that represents the updated `ccrz__E_Cart__c` record.

**Example: Example**

Apply a discount to a cart associated with a particular account, based on a value specified in a custom field, Account Discount, on the standard account object.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [subsc\\_cc\\_hk\\_PricingCartAcctDisc](#)



**Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.cc\\_bean\\_CartSummary](#)

[ccrz.ccLogicCartPrice](#)

## ccrz.cc\_api\_PriceAdjustment.computePricingReview


Evaluates a cart when the Order Review section of the Checkout page loads, and returns any applicable error messages to show to the buyer.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

 **Important:** This method defines legacy behavior, and by default the managed package instead references the methods of the `ccrz.ccLogicCartPrice` class.

## Signature

```
global virtual Map<String, Object> computePricingReview (Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_api_PriceAdjustment.PARAM_CART`

`ccrz__E_Cart__c` to evaluate.



**Warning:** Don't execute the DML `update` statement on this record within this method. The record is updated after the method completes.

### `ccrz.cc_api_PriceAdjustment.PARAM_CARTITEMS`

`List<ccrz__E_CartItem__c>` of the cart items included in the cart.



**Warning:** Don't execute the DML `update` statement on these records within this method. These records are updated after the method completes.

### `ccrz.cc_api_PriceAdjustment.PARAM_CURRENCYCODE`

String that specifies an ISO 4217 currency code, such as `USD` or `JPY`.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.userCurrency`.

### `ccrz.cc_api_PriceAdjustment.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### `ccrz.cc_api_PriceAdjustment.PARAM_USERID`

ID of the user assigned to the cart:

- When an authenticated buyer is logged in, this value is the ID of their user record.
- When a customer service representative (CSR) is logged in on behalf of a buyer, this value is the buyer's user ID.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_api_PriceAdjustment.PARAM_EXTRINSIC`

`Map<String, String>` that contains custom cart data to include in the `Handlebars.js` template that renders the Order Review section of the Checkout page. This information has the `extrinsic` attribute set.

**ccrz.cc\_excn\_BeanMessagesException**

`ccrz.cc_bean_Message` or `List<ccrz.cc_bean_Message>` that describe errors to show to the buyer on the Checkout page.

SEE ALSO:

[ccrz.ccLogicCartPrice](#)

[Inserting and Updating Records](#)

**ccrz.cc\_api\_PriceAdjustment.getInstance**

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_PriceAdjustment` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
global static ccrz.cc_api_PriceAdjustment getInstance (Map<String, Object>)
```

**Inputs (Optional)**

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_api\_PriceAdjustment.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

**Outputs**

This method returns an instance of `ccrz.cc_api_PriceAdjustment`, or your custom extension point class.

**ccrz.cc\_api\_ProductQuantityRule**

Define the quantity increments that a buyer can only use when adding or updating cart items. For example, maybe you want to require buyers to purchase a certain product only in increments of five.

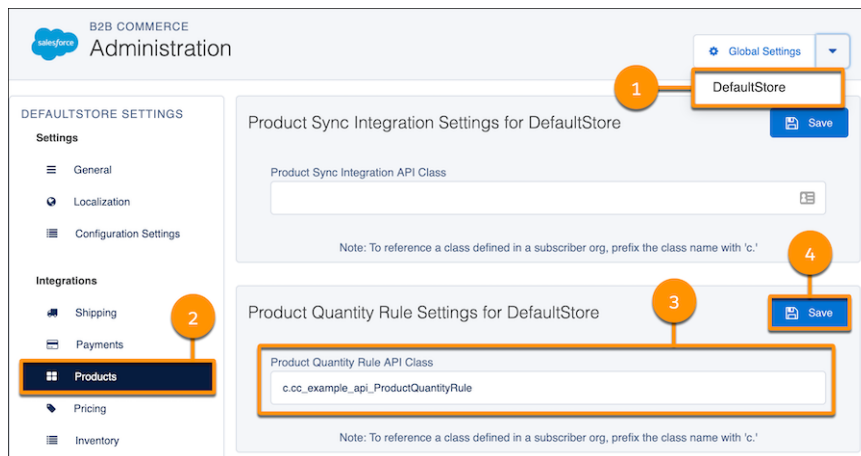
## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Products** in the Integrations section of the Storefront Settings menu (2). Update the Product Quantity Rule API Class field (3) with your custom class name, prefixed with `c.`, such as `c.cc_example_api_DeliveryDate`. Then, click **Save** (4).



**Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

**Tip:** The managed package includes some example extensions of the default `ccrz.cc_api_ProductQuantityRule` class:

- `fiveAndTen`, which specifies an Increment value of 5 and a Skip Increment value of 10 for all products. To enable this class in your storefront for a demo, enter `cc_api_ProductQuantityRule:fiveAndTen` for Product Quantity Rule API Class in your storefront settings.
- `single`, which specifies an Increment value of 1 but no Skip Increment value. To enable this class in your storefront for a demo, enter `cc_api_ProductQuantityRule:single` for Product Quantity Rule API Class in your storefront settings. This class is available only with [B2B Commerce for Visualforce Spring '20 \(version 4.12\) patch 3.147.8 or later](#).

## Front-End Input

When you extend this class to define quantity rules, the storefront provides increment controls on the Shopping Cart page. The storefront also shows increment controls for the Add to Cart or Add to Wish List button on other pages, such as the Product Detail page.

The Mini Cart and Quick Order widgets can also automatically change quantities to allowed increments, or show an error for an invalid quantity. For help with configuring the widget behavior, see the following help topics:

- [Show the Mini Cart Widget](#)
- [Show the Quick Order Widget](#)

## Methods

### IN THIS SECTION:

#### [getRules](#)

Returns quantity increment rules for a set of product SKUs.

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_ProductQuantityRule` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class. This method is a `global static` utility function and doesn't support overrides.

### SEE ALSO:

[Salesforce Help Enforce Quantity Rules When a Buyer Adds Products to a Cart or Wish List](#)

## ccrz.cc\_api\_ProductQuantityRule.getRules

Returns quantity increment rules for a set of product SKUs.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Map<String, Object>> getRules(List<String> SKUs)
```



## Inputs (Optional)

### SKUs

List<String> of specific product SKUs to define quantity increments for.

## Outputs

Map<String, Map<String, Object>>, keyed by SKU, where each Map<String, Object> can include the following keys:

### ccrz.cc\_api\_ProductQuantityRule.PARM\_INCREMENT

Integer that specifies the increment value that a buyer can increase or decrease a product's quantity by, such as 5. On the storefront, this value determines:

- The quantity that the Increment buttons apply.
- The quantity that the Mini Cart or Quick Order widget require or automatically set, depending on how you configure the widgets.

### ccrz.cc\_api\_ProductQuantityRule.PARM\_SKIP\_INCREMENT

Integer that specifies a larger increment value that a buyer can increase or decrease a product's quantity by, such as 10. On the storefront, this value determines the quantity that the Skip Increment buttons apply.

### ccrz.cc\_api\_ProductQuantityRule.PARM\_MISC\_DETAILS

String returned to the storefront, which you can customize and show to the buyer with some extra front-end customization.



**Tip:** For more information about Increment and Skip Increment buttons on the storefront, see [Enforce Quantity Rules When a Buyer Adds Products to a Cart or Wish List](#).



### Example: Example: Five and Ten

The managed package includes an example extension of the default `ccrz.cc_api_ProductQuantityRule` class. The extension, named `fiveAndTen`, specifies an Increment value of 5 and a Skip Increment value of 10 for all products. This class also returns the storefront name for `ccrz.cc_api_ProductQuantityRule.PARM_MISC_DETAILS`.

```
global class fiveAndTen extends ccrz.cc_api_ProductQuantityRule {
    private Map<String, Object> storefrontSettings;
    global fiveAndTen{}
    global override void setStorefrontSettings(New Map<String, Object> settings) {
        this.storeSettings = settings;
    }

    global override Map<String, Map<String, Object>> getRules(List<String> skus) {
        final String storefrontName = (String)storeSettings.get('Name');
        Map<String, Map<String, Object>> rulesOutputData = new Map<String, Map<String,
Object>>>;
        for (String sku:skus) {
            rulesOutputData.put(sku, new Map<String, Object>{
                PARM_INCREMENT => 5,
                PARM_SKIP_INCREMENT=>10,
                PARM_MISC_DETAILS=>'Storefront: ' + storefrontName}
            );
        }
        return rulesOutputData;
    }
}
```

### Example: Example: Single

In [B2B Commerce for Visualforce Spring '20 \(version 4.12\) patch 3.147.8 or later](#), the managed package includes another example extension of the default `ccrz.cc_api_ProductQuantityRule` class. The extension, named `single`, Increment value of 1 but no Skip Increment value.

```
public class single extends ccrz.cc_api_ProductQuantityRule {
    private Map<String, Object> storefrontSettings;
    public single() {}
    public override void setStorefrontSettings(Map<String, Object> settings) {
        this.storeSettings = settings;
    }

    public override Map<String, Map<String, Object>> getRules(List<String> skus) {
        final String storefrontName = (String)storeSettings.get('Name');
        Map<String, Map<String, Object>> rulesOutputData = new
Map<String, Map<String, Object>>();
        for(String sku:skus){
            rulesOutputData.put(sku, new Map<String, Object>{
                PARM_INCREMENT=>1,
                PARM_MISC_DETAILS=>'Storefront: ' + storefrontName}
            );
        }
        return rulesOutputData;
    }
}
```

### Example: Example: Reference Custom Fields on a Product

A subscriber has extended the `ccrz__E_Product__c` object to add custom fields that specify the Increment (`example__Increment__c`) and Skip Increment (`example__SkipIncrement__c`) values. This example shows an extension of the default `ccrz.cc_api_ProductQuantityRule` class that references these custom fields.

```
global class cc_example_api_ProductQuantityRule extends ccrz.cc_api_ProductQuantityRule{

    global override Map<String, Map<String, Object>> getRules(List<String> skus) {
        // Construct the return map
        Map<String, Map<String, Object>> rulesOutputData = new Map<String, Map<String,
Object>>;

        // Iterate over product records
        for(ccrz__E_Product__c prod: [
            SELECT example_Increment__c, example_SkipIncrement__c, ccrz__SKU__c
            FROM ccrz__E_Product__c
            WHERE ccrz__SKU__c IN :skus]){
            // For each product
            Map<String, Object> quantityRules = new Map<String, Object>;
            // Check whether the Increment and Skip Increment values have a value,
and populate the quantityRules map
            if(null != prod.example__Increment__c){
                quantityRules.put(ccrz.cc_api_ProductQuantityRule.PARM_INCREMENT,
prod.example__Increment__c.intValue);
            }
            if(null != prod.example__SkipIncrement__c){
```

```

quantityRules.put(ccrz.cc_api_ProductQuantityRule.PARM_SKIP_INCREMENT,
prod.example__SkipIncrement__c.intValue);
    }

    // Set the rules for the specific product
    rulesOutputData.put(prod.ccrz__SKU__c, quantityRules);
    }
    return rulesOutputData;
}
}

```

**SEE ALSO:**

[Salesforce Help Enforce Quantity Rules When a Buyer Adds Products to a Cart or Wish List](#)

[Salesforce Help Show the Mini Cart Widget](#)

[Salesforce Help Show the Quick Order Widget](#)

## ccrz.cc\_api\_ProductQuantityRule.getInstance

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_ProductQuantityRule` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class. This method is a `global static` utility function and doesn't support overrides.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global static ccrz.cc_api_ProductQuantityRule getInstance(Map<String, Object>)
```

### Inputs (Optional)

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_api\_ProductQuantityRule.PARAM\_STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_api_ProductQuantityRule`, or your custom extension point class.

## ccrz.cc\_api\_ShippingAndHandling

Implement subscriber-defined behavior for populating the available shipping methods during checkout. On the storefront, the Checkout page calls this extension point class before rendering shipping options.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Shipping** in the Integrations section of the Storefront Settings menu (2). Update the Shipping Integration API Class field (3) with your custom class name, prefixed with `c.`, such as `c.cc_example_api_ShippingAndHandling`.

The screenshot shows the B2B Commerce Administration interface. The top navigation bar includes 'Global Settings' and 'DefaultStore' (callout 1). The left sidebar shows 'Integrations' selected (callout 2). The main content area displays 'Shipping Integration Settings for DefaultStore' with a 'Shipping Integration API Class' field containing 'c.cc\_example\_api\_ShippingAndHandling' (callout 3). Below this is the 'Delivery Date Settings for DefaultStore' section with a 'Delivery Date API Class' field containing 'cc\_api\_DeliveryDate'. Both sections include a 'Save' button and a note: 'Note: To reference a class defined in a subscriber org, prefix the class name with 'c.''

 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

**Important:** The default implementation of this class is intended for demo purposes only, and not meant to be used in a production org. This class provides a starting point that you can extend for implementing the shipping options that are specific to your solution's unique requirements.

**Tip:** The managed package includes an example extension of the default `ccrz.cc_api_ShippingAndHandling` class. The extension, named `fixedShippingThreeOptions`, provides three flat-rate shipping options:

- Slow shipping for \$5.00
- Fast shipping for \$10.00
- Fastest shipping for \$20.00

## Front-End Input

On the storefront, a buyer selects a shipping method from the Shipping Method picklist on the Checkout page. For help with configuring the shipping options on the Checkout page, see [Specify Shipping Options for Standard Checkout](#).

## Methods

### IN THIS SECTION:

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_ShippingAndHandling` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

#### [getShippingOptions](#)

Returns available options for the Shipping Method picklist on the storefront's Checkout page. To implement shipping in your production solution, such as to integrate an external shipping provider, override this method with subscriber code.

### SEE ALSO:

[Salesforce Help Specify Shipping Options for Standard Checkout](#)

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

## ccrz.cc\_api\_ShippingAndHandling.getInstance

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_api_ShippingAndHandling` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static ccrz.cc_api_ShippingAndHandling getInstance(Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_api_ShippingAndHandling.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_api_ShippingAndHandling`, or your custom extension point class.

## `ccrz.cc_api_ShippingAndHandling.getShippingOptions`

Returns available options for the Shipping Method picklist on the storefront's Checkout page. To implement shipping in your production solution, such as to integrate an external shipping provider, override this method with subscriber code.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Important:** The default implementation of this method is intended for demo purposes only, and not meant to be used in a production org. This method provides a starting point that you can extend for implementing the shipping options that are specific to your solution's unique requirements.

## Signature

```
global virtual List<ccrz.cc_ctrl_hlpr_ShippingOption> getShippingOptions(String zipCode, String stateCode, String countryCode, ID cartID, String storeName)
```

## Inputs (Required)

### zipCode

String that specifies the postal code from the cart's shipping address.

### stateCode

String that specifies the ISO 3166-2 code for a US state or Canadian province from the cart's shipping address.

### countryCode

String that specifies the ISO 3166-1 code for the country or territory from the cart's shipping address.

### cartID

Salesforce ID (18 digits) of the cart used for checkout.

### storeName

String that specifies a storefront name.

## Outputs

List<[ccrz.cc\\_ctrl\\_hlpr\\_ShippingOption](#)> that represents the available shipping methods.



### Example: Examples

The managed package includes an example extension of the default `ccrz.cc_api_ShippingAndHandling` class. The extension, named `fixedShippingThreeOptions`, provides three flat-rate shipping options:

- Slow shipping for \$5.00
- Fast shipping for \$10.00
- Fastest shipping for \$20.00

```
global class fixedShippingThreeOptions extends ccrz.cc_api_ShippingAndHandling{
    global fixedShippingThreeOptions() {}
    global override List<ccrz.cc_ctrl_hlpr_ShippingOption> getShippingOptions(String
zipCode, String stateCode, String countryCode, Id cartId, String storeName) {
        return new <List<ccrz.cc_ctrl_hlpr_ShippingOption> {
            new cc_ctrl_hlpr_ShippingOption('Slow', 'Standard', 5.00),
            new cc_ctrl_hlpr_ShippingOption('Fast', 'Standard', 10.00),
            new cc_ctrl_hlpr_ShippingOption('Fastest', 'Standard', 20.00)
        };
    }
}
```

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

[ccrz.ccLogicCartGetShippingOptions](#)

[Salesforce Help Specify Shipping Options for Standard Checkout](#)

[Salesforce Help Specify Shipping Group Options for LLI Checkout](#)

## ccrz.cc\_hk\_Catalog

Return info from your product catalog used for populating autocomplete suggestions on the storefront.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Products** in the Integrations section of the Storefront Settings menu (2). Update the Catalog API Class field (3) with your custom class name, prefixed with `c.`, such as `c.cc_example_hk_Catalog`.

The screenshot displays the B2B Commerce Administration interface. At the top, it shows 'B2B COMMERCE Administration' and 'Global Settings' for the 'DefaultStore' storefront (callout 1). The left sidebar is divided into 'DEFAULTSTORE SETTINGS' and 'Integrations'. Under 'Integrations', the 'Products' option is selected (callout 2). The main content area shows three settings panels for the 'DefaultStore':

- Product Sync Integration Settings for DefaultStore:** Includes a field for 'Product Sync Integration API Class'.
- Product Quantity Rule Settings for DefaultStore:** Includes a field for 'Product Quantity Rule API Class' with the value 'cc\_api\_ProductQuantityRule'.
- Catalog Settings for DefaultStore:** Includes a field for 'Catalog API Class' with the value 'c.cc\_example\_hk\_Catalog' (callout 3).

Each panel includes a 'Save' button and a note: 'Note: To reference a class defined in a subscriber org, prefix the class name with 'c.''

 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).



## Methods

### IN THIS SECTION:

#### [filterCatalogData](#)

Returns a list of product SKUs, for use with only version 1.0 of the Product List page.

#### [autoComplete](#)

Queries product records for populating SKU autocomplete suggestions in the Quick Order widget or product autocomplete suggestions in the Search Box widget.

#### [modifyAutoComplete](#)

Provides an extension point for modifying the list of autocomplete suggestions that the `ccrz.cc_hk_Catalog.autoComplete` method returns. The default implementation of this method removes duplicate results from the list.

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Catalog` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## ccrz.cc\_hk\_Catalog.filterCatalogData

Returns a list of product SKUs, for use with only version 1.0 of the Product List page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9



**Note:** This method is used only with version 1.0 of the Product List page.

## Signature

```
global virtual Map<String, Object> filterCatalogData (Map<String, Object>)
```

## Inputs

`Map<String, Object>` that can include the following keys:

#### **ccrz.cc\_hk\_Catalog.PARAM\_SKU\_LIST**

`List<String>` of product SKUs to return.

## Outputs

Map<String, Object> that can include the following keys:

### **ccrz.cc\_hk\_Catalog.PARAM\_SKU\_LIST**

List<String> of product SKUs.

## ccrz.cc\_hk\_Catalog.autoComplete

Queries product records for populating SKU autocomplete suggestions in the Quick Order widget or product autocomplete suggestions in the Search Box widget.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> autoComplete (Map<String, Object>)
```

## Inputs

Map<String, Object> that can include the following keys:

### **ccrz.cc\_hk\_Catalog.PARAM\_RESULTS\_LIMIT**

Integer that specifies the maximum number of results to return.



**Note:** By default, this method doesn't evaluate this limit.

### **ccrz.cc\_hk\_Catalog.PARAM\_SCOPE\_STRING**

String that specifies which widget the autocomplete suggestions are for, and how the results are queried. Your version of the managed package and the version of the Product List page that you've enabled for your storefront also affect the query.

Value	Usage	Managed Package Version	Product List Page Version	Query for Results
ccrz.cc_hk_Catalog.PARAM_SCOPE_STRING	Product suggestions for the Search Box widget	Summer '19 (version 4.11) or later	2.0	SOSL query of <a href="#">ccrz__E_ProductIndex_c</a> records

Value	Usage	Managed Package Version	Product List Page Version	Query for Results
		Summer '19 (version 4.11) or later	1.0	SOQL query of <a href="#">ccrz__E_Product__c</a> records
		Spring '19 (version 4.10) or earlier	Any version	SOQL query of <a href="#">ccrz__E_Product__c</a> records
<a href="#">ccrz.cc_hk_Catalog.SEQUENTIAL</a>	SKU suggestions for the Quick Order widget	Any version	Any version	SOQL query of <a href="#">ccrz__E_Product__c</a> records

**ccrz.cc\_hk\_Catalog.PARAM\_SEARCH\_STRING**

String to provide autocomplete suggestions for.

**ccrz.cc\_hk\_Catalog.PARAM\_STOREFRONT\_STRING**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to [ccrz.cc\\_CallContext.storefront](#).

## Outputs

Map<String, Object> that can include the following keys:

**ccrz.cc\_hk\_Catalog.PARAM\_SEARCH\_RESULTS**

List<String> of autocomplete suggestions that match the input criteria.

SEE ALSO:

[Salesforce Help Considerations for Autocomplete Suggestions in Search Box and Quick Order Widgets](#)

[Salesforce Help Product List Page](#)

## ccrz.cc\_hk\_Catalog.modifyAutoComplete

Provides an extension point for modifying the list of autocomplete suggestions that the `ccrz.cc_hk_Catalog.autoComplete` method returns. The default implementation of this method removes duplicate results from the list.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> modifyAutoComplete (Map<String, Object>)
```

## Inputs

`Map<String, Object>` that can include the following keys:

**`ccrz.cc_hk_Catalog.PARAM_RESULTS_LIMIT`**

Integer that specifies the maximum number of results to return.

**`ccrz.cc_hk_Catalog.PARAM_SEARCH_RESULTS`**

`List<String>` of autocomplete suggestions that the `ccrz.cc_hk_Catalog.autoComplete` method returns.

## Outputs

`Map<String, Object>` that can include the following keys:

**`ccrz.cc_hk_Catalog.PARAM_SEARCH_RESULTS`**

`List<String>` of modified autocomplete suggestions.

## `ccrz.cc_hk_Catalog.getInstance`

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Catalog` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static ccrz.cc_hk_Catalog getInstance (Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_Catalog.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_hk_Catalog`, or your custom extension point class.

## ccrz.cc\_hk\_Category (Deprecated)

Dynamically construct the category tree for a storefront by querying cached category data.

## Compatibility

This class is deprecated. For existing extensions of this class, the managed package provides an example class, `ccrz.ccLogicCategoryGetTreeHk`, that extends the default `ccrz.ccLogicCategoryGetTree` logic service provider to reference methods of this class.

### Important:

- Use the `ccrz.ccLogicCategoryGetTreeHk` example for short-term compatibility with existing extensions of `ccrz.cc_hk_Category`. Migrate your extensions to an override of `ccrz.ccLogicCategoryGetTree`.
- For new solutions, extend `ccrz.ccLogicCategoryGetTree`.

## Methods

### IN THIS SECTION:

#### [fetch](#)

Defines legacy behavior for calling the `ccrz.ccApiPublicCache.fetch` method for returning the cached category tree data.

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Category` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

### SEE ALSO:

[ccrz.ccLogicCategoryGetTree](#)

[ccrz.ccLogicCategoryGetTreeHk Example Extension Class](#)

## ccrz.cc\_hk\_Category.fetch

Defines legacy behavior for calling the `ccrz.ccApiPublicCache.fetch` method for returning the cached category tree data.

## Compatibility

This class is deprecated. For existing extensions of this class, the managed package provides an example class, `ccrz.ccLogicCategoryGetTreeHk`, that extends the default `ccrz.ccLogicCategoryGetTree` logic service provider to reference methods of this class.

### Important:

- Use the `ccrz.ccLogicCategoryGetTreeHk` example for short-term compatibility with existing extensions of `ccrz.cc_hk_Category`. Migrate your extensions to `ccrz.ccLogicCategoryGetTree`.
- For new solutions, extend `ccrz.ccLogicCategoryGetTree`.

## Signature

```
global virtual Map<String, Object> fetch (Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_Category.PARAM_CATEGORY_ID`


String that specifies the 18-digit Salesforce ID of a category to fetch subcategories for.

## Outputs

`Map<String, Object>` that can include the following keys:


### `ccrz.cc_hk_Category.PARAM_CATEGORIES`

`List<Object>` of the cached category tree data, deserialized from the `ccrz.ccApiPublicCache.CONTENT` output of `ccrz.ccApiPublicCache.fetch`.

 **Note:** This output is returned only when the input data doesn't include `ccrz.cc_hk_Category.PARAM_CATEGORY_ID`.

### `ccrz.cc_hk_Category.PRAM_SUB_CAT`

`List<ccrz__E_Category__c>` of one level of subcategories for the specified `ccrz.cc_hk_Payment.PARAM_CATEGORY_ID`.

 **Note:**

- This output is returned only when the input data includes `ccrz.cc_hk_Category.PARAM_CATEGORY_ID`.
- In versions of CloudCraze earlier than 4.5, this output is a `List<cc_bean_Category>`.

### Example: Example

Override the default `ccrz.cc_hk_Category.fetch` method to rename the top-level categories.

```
global virtual override Map<String, Object> fetch (Map<String, Object> inputData {
    Map<String, Object> defaultFetchResults = super.fetch(inputData);

    List<Object> categories = (List<Object>)defaultFetchResults.get(PARAM_CATEGORIES);

    for(Object category:categoryTree) {
        Map<String, Object> categoryMap = (Map<String, Object>)category;
```

```

        categoryMap.put('name', 'stringToRename' + categoryMap.get('name'));
    }

    return getTreeResults;
}

```

SEE ALSO:

[ccrz.ccLogicCategoryGetTree](#)

[ccrz.ccLogicCategoryGetTreeHk Example Extension Class](#)

## ccrz.cc\_hk\_Category.getInstance

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Category` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

### Compatibility

This class is deprecated. For existing extensions of this class, the managed package provides an example class, [ccrz.ccLogicCategoryGetTreeHk](#), that extends the default [ccrz.ccLogicCategoryGetTree](#) logic service provider to reference methods of this class.

#### Important:

- Use the `ccrz.ccLogicCategoryGetTreeHk` example for short-term compatibility with existing extensions of `ccrz.cc_hk_Category`. Migrate your extensions to an override of `ccrz.ccLogicCategoryGetTree`.
- For new solutions, extend `ccrz.ccLogicCategoryGetTree`.

### Signature

```
global static ccrz.cc_hk_Order getInstance(Map<String, Object>)
```

### Inputs (Optional)

`Map<String, Object>` that can include the following keys:

#### **ccrz.cc\_hk\_Category.PARAM\_STOREFRONT**

String that specifies the name of a storefront.

 **Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### Outputs

This method returns an instance of `ccrz.cc_hk_Category`, or your custom extension point class.

## ccrz.cc\_hk\_DynamicTheme

Load a unique storefront theme based on the current buyer's account group.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Themes** in the Appearance section of the Storefront Settings menu (2). Update the Dynamic Theme Extension API Class field (3) with your custom class name, prefixed with *c.*, such as *c.cc\_example\_hk\_DynamicTheme*.

The screenshot shows the B2B Commerce Administration interface. On the left, the 'DEFAULTSTORE SETTINGS' sidebar is visible, with 'Appearance' expanded and 'Themes' selected (2). The main content area shows 'Themes for DefaultStore' with a table of themes and their actions. Below this, the 'Dynamic Theme Extension Settings for DefaultStore' section has a text field for 'Dynamic Theme Extension API Class' containing 'c.cc\_example\_hk\_DynamicTheme' (3). A note below the field states: 'Note: To reference a class defined in a subscriber org, prefix the class name with 'c.''. At the top right, a dropdown menu for 'DefaultStore' is open (1).

**Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).



## Methods

### IN THIS SECTION:

#### [determineTheme](#)

Returns the value of the Theme Name field on the current buyer's account group.

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_DynamicTheme` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

### SEE ALSO:

#### [ccrz\\_\\_E\\_AccountGroup\\_\\_c](#)

[Salesforce Help Show a Unique Storefront Theme for an Account Group](#)

[Salesforce Help Style Your Storefront with a Theme](#)

## ccrz.cc\_hk\_DynamicTheme.determineTheme

Returns the value of the Theme Name field on the current buyer's account group.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> determineTheme (Map<String, Object>)
```

## Inputs

None

## Outputs

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_hk\_DynamicTheme.PARAM\_THEME\_NAME**

String that specifies the name of the theme to load for the buyer's storefront session. The default implementation of this method references the value of the `ThemeName__c` field on the `ccrz__E_AccountGroup__c` record referenced by `ccrz.cc_CallContext.currAccountGroup`.

## SEE ALSO:

[ccrz\\_\\_E\\_AccountGroup\\_\\_c](#)

[Salesforce Help Show a Unique Storefront Theme for an Account Group](#)

[Salesforce Help Style Your Storefront with a Theme](#)

## ccrz.cc\_hk\_DynamicTheme.getInstance

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_DynamicTheme` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global static ccrz.cc_hk_DynamicTheme getInstance(Map<String, Object>)
```

### Inputs (Optional)

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_hk\_DynamicTheme.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

### Outputs

This method returns an instance of `ccrz.cc_hk_DynamicTheme`, or your custom extension point class.

## ccrz.cc\_hk\_EffectiveAccount

Return and filter effective accounts that a buyer can access other than the buyer's default assigned account. Update the active shopping cart for a selected effective account.

### Compatibility

This reference applies to:


Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Orders** in the Integrations section of the Storefront Settings menu (2). Update the Effective Account Hook API Class field (3) with your custom class name, prefixed with *c.*, such as *c.cc\_example\_hk\_EffectiveAccount*.

The screenshot shows the B2B Commerce Administration interface. The top navigation bar includes the Salesforce logo, 'B2B COMMERCE Administration', and a 'Global Settings' dropdown menu. The 'DefaultStore' dropdown is highlighted with a red circle and callout '1'. The left sidebar contains a 'Settings' section with sub-items: General, Localization, and Configuration Settings. Below this is an 'Integrations' section with sub-items: Shipping, Payments, Products, Pricing, Inventory, Orders (highlighted with a red circle and callout '2'), Carts, Tax, SSO, Promotions, and Service Management. The main content area displays three configuration sections for the 'DefaultStore':

- Outbound Orders Settings for DefaultStore:** Includes a text input field for 'Outbound Orders API Class' and a 'Save' button. A note below reads: 'Note: To reference a class defined in a subscriber org, prefix the class name with 'c.''
- Outbound Order Cancel Settings for DefaultStore:** Includes a text input field for 'Outbound Order Cancel API Class' with the value 'cc\_api\_OutboundOrderCancel' and a 'Save' button. A note below reads: 'Note: To reference a class defined in a subscriber org, prefix the class name with 'c.''
- Order Hook Settings for DefaultStore:** Includes a text input field for 'Order Hook API Class' with the value 'cc\_hk\_Order' and a 'Save' button. A note below reads: 'Note: To reference a class defined in a subscriber org, prefix the class name with 'c.''
- Effective Account Hook Settings for DefaultStore:** This section is highlighted with a red box and callout '3'. It includes a text input field for 'Effective Account Hook API Class' with the value 'c.cc\_example\_hk\_EffectiveAccount' and a 'Save' button. A note below reads: 'Note: To reference a class defined in a subscriber org, prefix the class name with 'c.''

 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

## Methods

### IN THIS SECTION:

#### [fetchAccounts](#)

Returns a list of effective accounts available for the buyer.

#### [buildEffectiveAccountData](#)

Constructs the `ccrz.cc_bean_EffectiveAccount` data within the `ccrz.cc_hk_EffectiveAccount.fetchAccounts` method. This method is a `global` utility function, and supports extension and overrides.

#### [filterAccounts](#)

Provides an extension point for extra filtering logic after the `ccrz.cc_hk_EffectiveAccount.fetchAccounts` method executes. You can override this method to implement custom filtering of the available effective accounts instead of overriding the entire `fetchAccounts` method.

#### [processDeliveryDates](#)

Calls the `ccrz.cc_api_DeliveryDate` extension point class for determining an effective account's available delivery dates. This method returns data only when the Process Delivery Date setting of the Effective Account configuration module is `TRUE` for your storefront.

[saveCart](#)

Updates certain fields on the active shopping cart record based on the selected effective account, and returns the saved cart record.

[getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_EffectiveAccount` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

[Extension Example for ccrz.cc\\_hk\\_EffectiveAccount](#)

To implement custom business logic for effective accounts, create custom subscriber code that extends the default `ccrz.cc_hk_EffectiveAccount` class. The following code shows examples of overriding the `fetchAccounts` and `saveCart` methods. This example relies on a Relationship custom object, which maps accounts available for shipping with accounts that complete storefront purchases.

## SEE ALSO:

[ccrz.cc\\_bean\\_EffectiveAccount](#)

[Salesforce Help Allow a Buyer to Purchase for Multiple Effective Accounts](#)

## ccrz.cc\_hk\_EffectiveAccount.fetchAccounts

Returns a list of effective accounts available for the buyer.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> fetchAccounts (Map<String, Object>)
```

### Inputs (Optional)

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_hk\_EffectiveAccount.PARAM\_ACCOUNT\_ID**

String that specifies the ID of the buyer's assigned account.

**ccrz.cc\_hk\_EffectiveAccount.PARAM\_FETCH\_MODE**

String that specifies the effective account strategy you're using.

Value	Usage
<i>FETCH_PARENT</i> (default)	The parent-child account hierarchy strategy.
<i>FETCH_ENTITLED</i>	The entitled strategy.
<i>FETCH_GROUP</i>	The account group strategy.

**ccrz.cc\_hk\_EffectiveAccount.PARAM\_ASSIGNED\_ACCT\_MODE**

String that specifies when to include the buyer's assigned account in the list of available effective accounts.

Value	Usage
<i>ASSIGNED_ACCOUNT_ALWAYS</i>	Always include the buyer's assigned account. For the account group strategy, the buyer's assigned account is always included.
<i>ASSIGNED_ACCOUNT_NEVER</i>	Never include the buyer's assigned account.
<i>ASSIGNED_ACCOUNT_ONLYEMPTY</i> (default)	Include the buyer's assigned account only when the buyer doesn't have sharing access to any other accounts.

**ccrz.cc\_hk\_EffectiveAccount.PARAM\_ADDR\_SOURCE**

String that specifies which record type provides the shipping addresses that appear for each effective account.

Value	Usage
<i>AddressBook</i>	The shipping address that's selected as the <i>default</i> shipping address from the effective account's address book. This address comes from the effective account's related <a href="#">ccrz__E_AccountAddressBook__c</a> and <a href="#">ccrz__E_ContactAddr__c</a> records.
<i>Account</i> (default)	The shipping address specified directly on the effective account's account record.

## Outputs

Map<[String](#), [Object](#)> that can include the following keys:

**ccrz.cc\_hk\_EffectiveAccount.PARAM\_EFFACCOUNT\_LIST**

List<[ccrz.cc\\_bean\\_EffectiveAccount](#)> of the effective accounts that the buyer can access.

SEE ALSO:

[ccrz.cc\\_bean\\_EffectiveAccount](#)

[Salesforce Help Allow a Buyer to Purchase for Multiple Effective Accounts](#)

[Salesforce Help Include the Buyer's Assigned Account in the Effective Account Selector](#)

## ccrz.cc\_hk\_EffectiveAccount.buildEffectiveAccountData

Constructs the `ccrz.cc_bean_EffectiveAccount` data within the `ccrz.cc_hk_EffectiveAccount.fetchAccounts` method. This method is a `global` utility function, and supports extension and overrides.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global List<ccrz.cc_bean_EffectiveAccount> buildEffectiveAccountData(List<Account>, String)
```

### Inputs (Optional)

Unlike most other methods of the B2B Commerce for Visualforce API, this utility method doesn't accept a `Map<String, Object>` of input data. Instead, this method accepts the following specific input parameters:

#### **accountList**

`List<Account>` of account records to construct `ccrz.cc_bean_EffectiveAccount` data for.

#### **ccrz.cc\_hk\_EffectiveAccount.PARAM\_ADDR\_SOURCE**

String that specifies which record type provides the shipping addresses that appear for each effective account.

Value	Usage
<i>AddressBook</i>	The shipping address that's selected as the <i>default</i> shipping address from the effective account's address book. This address comes from the effective account's related <code>ccrz__E_AccountAddressBook__c</code> and <code>ccrz__E_ContactAddr__c</code> records.
<i>Account</i> (default)	The shipping address specified directly on the effective account's account record.

## Outputs

This method returns a `List<ccrz.cc_bean_EffectiveAccount>` of the effective accounts that the buyer can access.

SEE ALSO:

[ccrz.cc\\_bean\\_EffectiveAccount](#)

[Salesforce Help Allow a Buyer to Purchase for Multiple Effective Accounts](#)

## ccrz.cc\_hk\_EffectiveAccount.filterAccounts

Provides an extension point for extra filtering logic after the `ccrz.cc_hk_EffectiveAccount.fetchAccounts` method executes. You can override this method to implement custom filtering of the available effective accounts instead of overriding the entire `fetchAccounts` method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> filterAccounts (Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_hk\_EffectiveAccount.PARAM\_ACCOUNT\_ID**

String that specifies the ID of the buyer's assigned account.

**ccrz.cc\_hk\_EffectiveAccount.PARAM\_ADDR\_SOURCE**

String that specifies which record type provides the shipping addresses that appear for each effective account.

Value	Usage
<i>AddressBook</i>	The shipping address that's selected as the <i>default</i> shipping address from the effective account's address book. This address comes from the effective account's related <a href="#">ccrz__E_AccountAddressBook__c</a> and <a href="#">ccrz__E_ContactAddr__c</a> records.
<i>Account</i>	The shipping address specified directly on the effective account's account record.



**ccrz.cc\_hk\_EffectiveAccount.PARAM\_ASSIGNED\_ACCT\_MODE**

String that specifies when to include the buyer's assigned account in the list of available effective accounts.

Value	Usage
<i>ASSIGNED_ACCOUNT_ALWAYS</i>	Always include the buyer's assigned account. For the account group strategy, the buyer's assigned account is always included.
<i>ASSIGNED_ACCOUNT_NEVER</i>	Never include the buyer's assigned account.
<i>ASSIGNED_ACCOUNT_ONLYEMPTY</i>	Include the buyer's assigned account only when the buyer doesn't have sharing access to any other accounts.

**ccrz.cc\_hk\_EffectiveAccount.PARAM\_FETCH\_MODE**

String that specifies the effective account strategy you're using.

Value	Usage
<i>FETCH_PARENT</i>	The parent-child account hierarchy strategy. This strategy is selected by default if you don't specify a different strategy.
<i>FETCH_ENTITLED</i>	The entitled strategy.
<i>FETCH_GROUP</i>	The account group strategy.

## Outputs

Map<[String](#), [Object](#)> that can include the following keys:

**ccrz.cc\_hk\_EffectiveAccount.PARAM\_EFFACCOUNT\_LIST**

List<[ccrz.cc\\_bean\\_EffectiveAccount](#)> of the effective accounts that the buyer can access.

SEE ALSO:

[ccrz.cc\\_bean\\_EffectiveAccount](#)

[Salesforce Help Allow a Buyer to Purchase for Multiple Effective Accounts](#)

## ccrz.cc\_hk\_EffectiveAccount.processDeliveryDates

Calls the `ccrz.cc_api_DeliveryDate` extension point class for determining an effective account's available delivery dates. This method returns data only when the Process Delivery Date setting of the Effective Account configuration module is `TRUE` for your storefront.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processDeliveryDates (Map<String, Object>)
```

## Inputs (Required)

This method accepts a `Map<String, Object>`, which contains input data to pass to the `ccrz.cc_api_DeliveryDate` extension point class.

## Outputs

This method returns a `Map<String, Object>` of delivery date data from the `ccrz.cc_api_DeliveryDate.getDeliveryDates` method.

SEE ALSO:

[ccrz.cc\\_api\\_DeliveryDate](#)

[Salesforce Help Manage Carts for Effective Accounts](#)

## ccrz.cc\_hk\_EffectiveAccount.saveCart

Updates certain fields on the active shopping cart record based on the selected effective account, and returns the saved cart record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> saveCart (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.cc_hk_EffectiveAccount.PARAM_EFFACCOUNT`

`ccrz.cc_bean_EffectiveAccount` that represents the buyer's selected effective account.

### `ccrz.cc_hk_EffectiveAccount.PARAM_CART_DATA`

`ccrz__E_Cart__c` that represents the active cart for the selected effective account.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_EffectiveAccount.PARAM_CART_DATA`

`ccrz__E_Cart__c` that represents the updated cart record. This method updates the following fields on the cart:

Field Name	Updated Value
<code>ccrz__BillTo__c</code>	A clone of the effective account's default billing address.
<code>ccrz__EffectiveAccountID__c</code>	ID of the account record that corresponds to the selected effective account.
<code>ccrz__RequestDate__c</code>	The selected or default delivery date returned from the <a href="#">ccrz.cc_api_DeliveryDate</a> extension point class, if applicable.
<code>ccrz__ShipTo__c</code>	A clone of the effective account's default shipping address.

SEE ALSO:

[Salesforce Help Manage Carts for Effective Accounts](#)

## `ccrz.cc_hk_EffectiveAccount.getInstance`

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_EffectiveAccount` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static ccrz.cc_hk_EffectiveAccount getInstance(Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_EffectiveAccount.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_hk_EffectiveAccount`, or your custom extension point class.

## Extension Example for `ccrz.cc_hk_EffectiveAccount`

To implement custom business logic for effective accounts, create custom subscriber code that extends the default `ccrz.cc_hk_EffectiveAccount` class. The following code shows examples of overriding the `fetchAccounts` and `saveCart` methods. This example relies on a Relationship custom object, which maps accounts available for shipping with accounts that complete storefront purchases.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.cc\\_sample\\_hk\\_EffectiveAccount](#)

**Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[ccrz.cc\\_hk\\_EffectiveAccount.fetchAccounts](#)

[ccrz.cc\\_hk\\_EffectiveAccount.saveCart](#)

## `ccrz.cc_hk_Invoice`

Create and return data for an invoice after a buyer completes a storefront purchase, and process payments associated with invoices.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.cc\\_hk\\_Invoice](#)

**Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

# ccrz.cc\_hk\_Menu

Dynamically construct the menu hierarchy for a storefront's navigation bar by querying cached menu data.

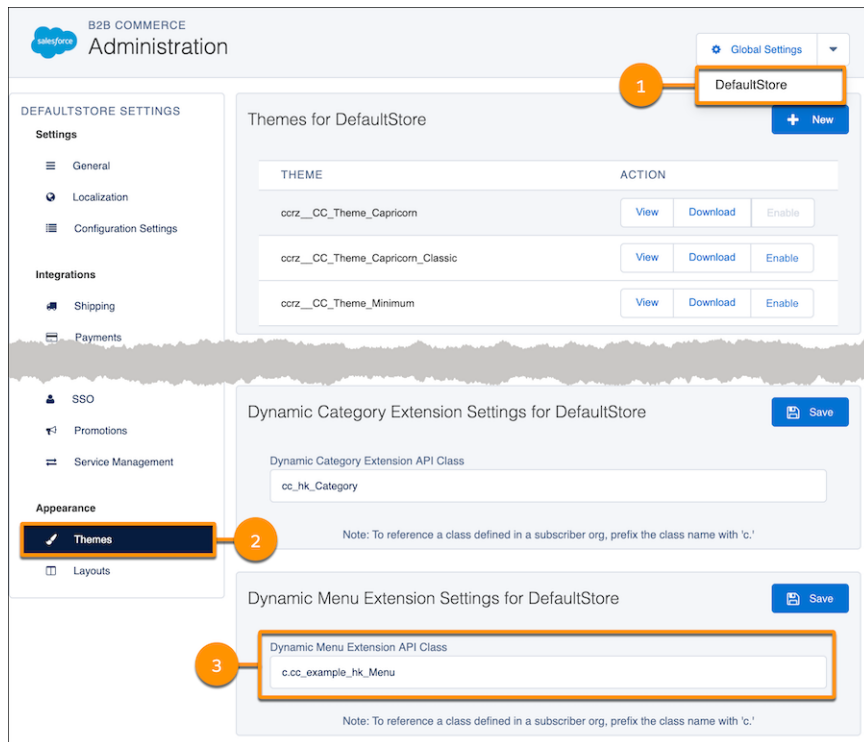
## Compatibility


This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Themes** in the Appearance section of the Storefront Settings menu (2). Update the Dynamic Menu Extension API Class field (3) with your custom class name, prefixed with *c.*, such as *c.cc\_example\_hk\_Menu*.



 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

## Methods

### IN THIS SECTION:

#### [fetch](#)

Calls the `ccrz.ccApiPublicCache.fetch` method to query the menu hierarchy from the cached menu data.

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Menu` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## ccrz.cc\_hk\_Menu.fetch

Calls the `ccrz.ccApiPublicCache.fetch` method to query the menu hierarchy from the cached menu data.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetch (Map<String, Object>)
```

## Inputs

None. This method relies on `ccrz.cc_CallContext` fields for information relevant for the fetch, such as storefront, account, account group, and so on.

## Outputs


`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_Menu.PARAM_MENU`

`List<ccrz.cc_bean_MenuItem>` of the storefront's top-level menu records. If a menu contains any nested menu items, the `children` field specifies another `List<ccrz.cc_bean_MenuItem>` of the nested menu items.

### Example: Examples

Return only the first menu from the cached menu data.

 **Tip:** To extend the default behavior and query cached menu data before you add custom filtering, [invoke the `super` keyword](#) in your method override.

```
global class cc_example_hk_Menu extends ccrz.cc_hk_Menu{
    global override Map<String, Object> fetch(Map<String, Object> menuInputData) {
        List<ccrz.cc_bean_MenuItem> menuReturnData =
        (List<ccrz.cc_bean_MenuItem>)super.fetch(menuInputData).get(ccrz.cc_hk_Menu.PARAM_MENU);

        return
            new Map<String, Object>{
                ccrz.cc_hk_Menu.PARAM_MENU => new
                List<ccrz.cc_bean_MenuItem>{menuReturnData.get(0)}
            };
    }
}
```

Modify the Display Name of all top-level menus to include a custom prefix, such as *Acme*.

```
global class cc_example_hk_Menu extends ccrz.cc_hk_Menu{
    global override Map<String, Object> fetch(Map<String, Object> menuInputData) {
        Map<String, Object> defaultMenuReturnData => super.fetch(menuInputData);
        List<ccrz.cc_bean_MenuItem> renamedMenuReturnData =
        (List<ccrz.cc_bean_MenuItem>)defaultMenuReturnData.get(ccrz.cc_hk_Menu.PARAM_MENU);
        for(ccrz.cc_bean_MenuItem menu:renamedMenuReturnData){
            menu.displayName = 'Acme ' + menu.displayName;
        }
        return renamedMenuReturnData;
    }
}
```

SEE ALSO:

[ccrz.ccApiPublicCache.fetch](#)

## ccrz.cc\_hk\_Menu.getInstance

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Menu` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static ccrz.cc_hk_Menu getInstance (Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_Menu.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_hk_Menu`, or your custom extension point class.

## ccrz.cc\_hk\_Order

Implement subscriber-defined behavior for placing an order, reordering, and showing order history. This class defines a mixture of legacy and current behavior, and the order logic service provider classes replace the legacy functionality from this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

In these versions of the managed package, the following logic service provider classes replace some methods of this extension point class:



Legacy Method	Replacement Class
<code>ccrz.cc_hk_Order.createTransaction</code>	None. This functionality remains current in the managed package.
<code>ccrz.cc_hk_Order.fetchOrderHistory</code>	<a href="#">ccrz.ccLogicOrderHistory</a>
<code>ccrz.cc_hk_Order.place</code>	<ul style="list-style-type: none"> <li>B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12) or later—<a href="#">ccrz.ccLogicCartPlace</a></li> <li>B2B Commerce for Visualforce Spring '21 (managed package version 4.12 and API version 11) or earlier—This functionality remains current in the managed package.</li> </ul>
<code>ccrz.cc_hk_Order.placeTarget</code>	None. This functionality remains current in the managed package.
<code>ccrz.cc_hk_Order.reorder</code>	<a href="#">ccrz.ccLogicOrderReOrder</a>

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Orders** in the Integrations section of the Storefront Settings menu (2). Update the Order Hook API Class field (3) with your custom class name, prefixed with `c.`, such as `c.cc_example_hk_Order`.

The screenshot shows the B2B Commerce Administration interface. The top navigation bar includes 'Global Settings' and 'DefaultStore' (marked with a circled 1). The left sidebar shows 'DEFAULTSTORE SETTINGS' with a sub-section 'Integrations' where 'Orders' is selected (marked with a circled 2). The main content area displays 'Order Hook Settings for DefaultStore' with a 'Save' button. The 'Order Hook API Class' field is populated with 'c.cc\_example\_hk\_Order' (marked with a circled 3). Below this field is a note: 'Note: To reference a class defined in a subscriber org, prefix the class name with 'c.''. Other sections visible include 'Outbound Orders Settings for DefaultStore' and 'Outbound Order Cancel Settings for DefaultStore', both with 'Save' buttons and similar notes.

 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

## Methods

### IN THIS SECTION:

#### [createTransaction](#)

Creates a `ccrz__E_TransactionPayment__c` record from transaction payment data passed from the `ccrz.cc_hk_Payment.processPayment` method.

#### [fetchOrderHistory](#)

Defines legacy behavior for populating the list of orders in the My Orders section of the My Account page. This method also supports search functionality of the order history.

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Order` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

#### [place](#)

Specifies details for converting a cart to an order throughout the checkout process. The `ccrz.cc_api_Cart.placeEOrder` public method calls this method multiple times from the same instance of `ccrz.cc_hk_Order`. Each `place` call requires specific inputs for the context of the call within the overall checkout process.

#### [placeTarget](#)

Appends an order query string parameter to the URL for the storefront's Order Confirmation page. To fully customize a buyer's navigation path when they complete checkout, override this method in your extension subscriber code.

#### [reorder](#)

Defines legacy behavior for copying the data for a completed order and its order items to a new cart and cart items.

## `ccrz.cc_hk_Order.createTransaction`

Creates a `ccrz__E_TransactionPayment__c` record from transaction payment data passed from the `ccrz.cc_hk_Payment.processPayment` method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createTransaction(Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**`ccrz.cc_hk_Payment.PARAM_ORDER_ID`**

String that specifies the 18-digit Salesforce ID of the order.

**`ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA`**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object that represents transaction payment properties and values that the `ccrz.cc_hk_Payment.processPayment` method returns.

## Outputs

`Map<String, Object>` that can include the following keys:

**`ccrz.cc_hk_Payment.PARAM_TRANSACTION_CREATED`**

`ccrz__E_TransactionPayment__c` that represents the created transaction payment record.

SEE ALSO:

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

[ccrz.cc\\_hk\\_Payment.processPayment](#)


## `ccrz.cc_hk_Order.fetchOrderHistory`

Defines legacy behavior for populating the list of orders in the My Orders section of the My Account page. This method also supports search functionality of the order history.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Important:** This method defines legacy behavior, and by default the managed package instead references the `ccrz.ccApiOrder.fetchOrderHistory` method. To reference this legacy method in your solution, [set the API Fetch Orders configuration of the My Orders module in your storefront configuration settings to `FALSE`](#). However, we recommend that you set this configuration to `FALSE` only for compatibility with existing extensions. For new solutions, leave this configuration set to `TRUE`. To implement custom order history behavior in new solutions, override the `ccrz.ccLogicOrderHistory.fetchHistory` logic service provider method instead.

## Signature

```
global virtual Map<String, Object> fetchOrderHistory(Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

**`ccrz.cc_hk_Order.PARAM_SEARCH_FROM_DATE`**

String that specifies a locale-specific date format that represents the start date for filtering order history results.

**`ccrz.cc_hk_Order.PARAM_SEARCH_TO_DATE`**

String that specifies a locale-specific date format that represents the end date for filtering order history results.

**`ccrz.cc_hk_Order.PARAM_SEARCH_ORDER_INPUT`**

String to search the account's order history for.

**`ccrz.cc_hk_Order.PARAM_SEARCH_ORDER_STATUS`**

String that specifies an order status to filter order history by.

## Outputs

`Map<String, Object>` that can include the following keys:

**`ccrz.cc_hk_Order.PARAM_ORDERS`**

`List<ccrz.cc_bean_MockOrder>` of order record data that match the search criteria.

SEE ALSO:

[ccrz.ccApiOrder.fetchOrderHistory](#)

[ccrz.ccLogicOrderHistory](#)

[Salesforce Help Allow a Buyer to Manage Their Orders on the My Account Page](#)

## `ccrz.cc_hk_Order.getInstance`

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Order` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static ccrz.cc_hk_Order getInstance(Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_Order.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_hk_Order`, or your custom extension point class.

## `ccrz.cc_hk_Order.place`

Specifies details for converting a cart to an order throughout the checkout process. The `ccrz.cc_api_Cart.placeEOrder` public method calls this method multiple times from the same instance of `ccrz.cc_hk_Order`. Each `place` call requires specific inputs for the context of the call within the overall checkout process.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> place(Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that contains keys specific to the context of the current `place` call in the checkout process. The value of the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` key determines which other inputs the map requires.

For example, when the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is the constant `ccrz.cc_hk_Order.STEP_START`, the `place` method's input map requires the following values:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_START,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID
});
```

#### **ccrz.cc\_hk\_Order.PARAM\_PLACE\_STEP**

String that specifies the current step of the place process, which determines what other keys the `place` method's input map requires. This key specifies one of the following constant values, in order of execution within

`ccrz.cc_api_Cart.placeEOrder`:

1. [ccrz.cc\\_hk\\_Order.STEP\\_START](#) on page 1347—Called at the beginning of the process.
2. [ccrz.cc\\_hk\\_Order.STEP\\_FETCH\\_CART](#) on page 1348—Called after fetching the cart.
3. [ccrz.cc\\_hk\\_Order.STEP\\_FETCH\\_COUPONS](#) on page 1348—Called after fetching any coupons applied to the cart.
4. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ADDRESSES\\_PRE](#) on page 1349—Called before inserting the order's billing and shipping addresses.
5. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ADDRESSES\\_POST](#) on page 1349—Called after inserting the order's billing and shipping addresses.
6. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_PRE](#) on page 1350—Called before inserting the new order record.
7. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_POST](#) on page 1350—Called after inserting the new order record.
8. [ccrz.cc\\_hk\\_Order.STEP\\_FETCH\\_CART\\_ITEMS](#) on page 1351—Called after querying for the cart's major line items, such as standard products, composite products, or coupons.
9. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_ITEMS\\_PRE](#) on page 1351—Called before inserting the new order's items that correspond to the cart's major line items.
10. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_ITEMS\\_POST](#) on page 1352—Called after inserting the new order's items that correspond to the cart's major line items.
11. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_ITEM\\_TIERS\\_PRE](#) on page 1353—Called before inserting the new order items' pricing tiers.
12. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_ITEM\\_TIERS\\_POST](#) on page 1354—Called after inserting the new order items' pricing tiers.
13. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_CHILD\\_ITEMS\\_PRE](#) on page 1354—Called before inserting the new order's items that correspond to the cart's minor line items, such as component products.
14. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_CHILD\\_ITEMS\\_POST](#) on page 1355—Called after inserting the new order's items that correspond to the cart's minor line items, such as component products.
15. [ccrz.cc\\_hk\\_Order.STEP\\_CART\\_CLOSE\\_PRE](#) on page 1356—Called after setting the cart's `ccrz__CartStatus__c` field to `Closed`, but before updating the `ccrz__E_Cart__c` record. This step also sets the cart's `ccrz__ActiveCart__c` field to `FALSE`.
16. [ccrz.cc\\_hk\\_Order.STEP\\_CART\\_CLOSE\\_POST](#) on page 1357—Called after setting the cart's `ccrz__CartStatus__c` field to `Closed` and after updating the `ccrz__E_Cart__c` record.
17. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_TERMS\\_PRE](#) on page 1358—Called before inserting the order terms, which the buyer acknowledged and accepted during checkout.
18. [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_ORDER\\_TERMS\\_POST](#) on page 1359—Called after inserting the order terms.

19. `ccrz.cc_hk_Order.STEP_CREATE_STORED_PAYMENT_PRE` on page 1360—Called before creating a stored payment from the payment details specified during checkout.
20. `ccrz.cc_hk_Order.STEP_CREATE_STORED_PAYMENT_POST` on page 1361—Called after creating a stored payment from the payment details specified during checkout.
21. `ccrz.cc_hk_Order.STEP_CREATE_SUBSCRIPTION_PRE` on page 1362—Called before creating a subscription that's included in the order.
22. `ccrz.cc_hk_Order.STEP_CREATE_SUBSCRIPTION_POST` on page 1363—Called after creating a subscription that's included in the order.
23. `ccrz.cc_hk_Order.STEP_CREATE_TRANSACTION_PRE` on page 1364—Called before creating a transaction payment record from the payment details specified during checkout.
24. `ccrz.cc_hk_Order.STEP_CREATE_TRANSACTION_POST` on page 1365—Called after creating a transaction payment record from the payment details specified during checkout.
25. `ccrz.cc_hk_Order.STEP_PAYMENT_PROCESS_PRE` on page 1366—Called before processing the payment.
26. `ccrz.cc_hk_Order.STEP_PAYMENT_PROCESS_POST` on page 1367—Called after processing the payment.
27. `ccrz.cc_hk_Order.STEP_UPDATE_ORDER_STATUS_PRE` on page 1369—Called before updating the status of the order to submitted.
28. `ccrz.cc_hk_Order.STEP_UPDATE_ORDER_STATUS_POST` on page 1370—Called after updating the status of the order.
29. `ccrz.cc_hk_Order.STEP_INVOICE_CREATE_PRE` on page 1371—Called before creating an invoice.
30. `ccrz.cc_hk_Order.STEP_INVOICE_CREATE_POST` on page 1372—Called after creating an invoice.
31. `ccrz.cc_hk_Order.STEP_END` on page 1374—Called at the end of the process.

### Other Inputs Required for `ccrz.cc_hk_Order.STEP_START`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_START`, the `place` method's input map requires the following other inputs:


```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_START,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID
});
```

#### `ccrz.cc_hk_Order.PARAM_CART_ID`

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

#### `ccrz.cc_hk_Order.PARAM_ENC_CART_ID`

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

-  **Note:** This key is deprecated in new solutions, but the default checkout implementation continues to pass this key for compatibility with legacy code. If you override this method in a new solution, use only the `ccrz.cc_hk_Order.PARAM_CART_ID` input.


### Other Inputs Required for `ccrz.cc_hk_Order.STEP_FETCH_CART`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_FETCH_CART`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_FETCH_CART,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID
});
```

#### `ccrz.cc_hk_Order.PARAM_CART`

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

#### `ccrz.cc_hk_Order.PARAM_CART_ID`

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

#### `ccrz.cc_hk_Order.PARAM_ENC_CART_ID`

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.


### Other Inputs Required for `ccrz.cc_hk_Order.STEP_FETCH_COUPONS`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_FETCH_COUPONS`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_FETCH_COUPONS,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_COUPONS => coupon list
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID
});
```

#### `ccrz.cc_hk_Order.PARAM_CART`

`ccrz__E_Cart__c` record used for placing the order.


 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

#### `ccrz.cc_hk_Order.PARAM_CART_ID`

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

#### `ccrz.cc_hk_Order.PARAM_COUPONS`

List<`ccrz__E_Coupon__c`> of coupon records associated with the cart.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Coupon__c` custom object. If you've added any custom subscriber fields, query those fields separately.

#### `ccrz.cc_hk_Order.PARAM_ENC_CART_ID`

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.



## Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_ADDRESSES_PRE`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_ADDRESSES_PRE`, the `place` method's input map requires the following other inputs:


```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ADDRESSES_PRE,
    ccrz.cc_hk_Order.PARAM_BILLING_ADDR => billing address,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_SHIPPING_ADDR => shipping address
});
```

### `ccrz.cc_hk_Order.PARAM_BILLING_ADDR`

`ccrz__E_ContactAddr__c` record for the order's billing address.

### `ccrz.cc_hk_Order.PARAM_CART`

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

### `ccrz.cc_hk_Order.PARAM_CART_ID`

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_ENC_CART_ID`

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_SHIPPING_ADDR`

`ccrz__E_ContactAddr__c` record for the order's shipping address.

## Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_ADDRESSES_POST`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_ADDRESSES_POST`, the `place` method's input map requires the following other inputs:


```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ADDRESSES_POST,
    ccrz.cc_hk_Order.PARAM_BILLING_ADDR => billing address,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_SHIPPING_ADDR => shipping address
});
```

### `ccrz.cc_hk_Order.PARAM_BILLING_ADDR`

`ccrz__E_ContactAddr__c` record for the order's billing address.

### `ccrz.cc_hk_Order.PARAM_CART`

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_SHIPPING\_ADDR**

`ccrz__E_ContactAddr__c` record for the order's shipping address.


**Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_ORDER_PRE`**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_ORDER_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_PRE,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record ready to insert.


**Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_ORDER_POST`**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_ORDER_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_POST,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.


**Other Inputs Required for `ccrz.cc_hk_Order.STEP_FETCH_CART_ITEMS`**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_FETCH_CART_ITEMS`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_FETCH_CART_ITEMS,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEMS_PRE`**


When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEMS_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEMS_PRE,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
});
```

```
ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items to insert and that correspond to the cart's major line items, such as standard products, composite products, or coupons.


**Other Inputs Required for ccrz.cc\_hk\_Order.STEP\_CREATE\_ORDER\_ITEMS\_POST**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEMS_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEMS_POST,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**Other Inputs Required for ccrz.cc\_hk\_Order.STEP\_CREATE\_ORDER\_ITEM\_TIERS\_PRE**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is

`ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEM_TIERS_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEM_TIERS_PRE,

    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.



**Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers to insert.

## Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEM_TIERS_POST`


When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEM_TIERS_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_ITEM_TIERS_POST,

    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers
});
```

### `ccrz.cc_hk_Order.PARAM_CART`

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

### `ccrz.cc_hk_Order.PARAM_CART_ID`

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_CART_ITEMS`

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

### `ccrz.cc_hk_Order.PARAM_ENC_CART_ID`

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_ORDER`

`ccrz__E_Order__c` record that was inserted.

### `ccrz.cc_hk_Order.PARAM_ORDER_ITEMS`

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

### `ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS`

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

## Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_ORDER_CHILD_ITEMS_PRE`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_ORDER_CHILD_ITEMS_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_CHILD_ITEMS_PRE,

    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
```


```

ccrz.cc_hk_Order.PARAM_ORDER => order,
ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers
});

```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items to insert and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

**Other Inputs Required for ccrz.cc\_hk\_Order.STEP\_CREATE\_ORDER\_CHILD\_ITEMS\_POST**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is

`ccrz.cc_hk_Order.STEP_CREATE_ORDER_CHILD_ITEMS_POST`, the `place` method's input map requires the following other inputs:

```


ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_CHILD_ITEMS_POST,

    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers
});

```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.


## Other Inputs Required for `ccrz.cc_hk_Order.STEP_CART_CLOSE_PRE`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CART_CLOSE_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CART_CLOSE_PRE,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.



**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.


**Other Inputs Required for `ccrz.cc_hk_Order.STEP_CART_CLOSE_POST`**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CART_CLOSE_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CART_CLOSE_POST,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<ccrz\_\_E\_OrderItemPricingTier\_\_c> of the order items' pricing tiers that were inserted.

**Other Inputs Required for ccrz.cc\_hk\_Order.STEP\_CREATE\_ORDER\_TERMS\_PRE**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is

`ccrz.cc_hk_Order.STEP_CREATE_ORDER_TERMS_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_TERMS_PRE,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.



**Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<ccrz\_\_E\_CartItem\_\_c> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<ccrz\_\_E\_OrderItem\_\_c> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<ccrz\_\_E\_OrderItem\_\_c> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<ccrz\_\_E\_OrderItemPricingTier\_\_c> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<ccrz\_\_E\_OrderTerm\_\_c> of the order terms ready to insert.


## Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_ORDER_TERMS_POST`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_ORDER_TERMS_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_ORDER_TERMS_POST,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms
});
```

### `ccrz.cc_hk_Order.PARAM_CART`

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

### `ccrz.cc_hk_Order.PARAM_CART_ID`

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_CART_ITEMS`

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

### `ccrz.cc_hk_Order.PARAM_ENC_CART_ID`

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_ORDER`

`ccrz__E_Order__c` record that was inserted.

### `ccrz.cc_hk_Order.PARAM_ORDER_ITEMS`

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

### `ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS`

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

### `ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS`

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

### `ccrz.cc_hk_Order.PARAM_ORDER_TERMS`

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

## Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_STORED_PAYMENT_PRE`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_STORED_PAYMENT_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_STORED_PAYMENT_PRE,

    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment data
});
```

### `ccrz.cc_hk_Order.PARAM_CART`

`ccrz__E_Cart__c` record used for placing the order.



**Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

### `ccrz.cc_hk_Order.PARAM_CART_ID`

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_CART_ITEMS`

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

### `ccrz.cc_hk_Order.PARAM_ENC_CART_ID`

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_ORDER`

`ccrz__E_Order__c` record that was inserted.

### `ccrz.cc_hk_Order.PARAM_ORDER_ITEMS`

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

### `ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS`

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

### `ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS`

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

### `ccrz.cc_hk_Order.PARAM_ORDER_TERMS`

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values to pass to [ccrz.cc\\_hk\\_Payment.createStoredPayment](#).

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

[ccrz.cc\\_hk\\_Payment.TransactionPaymentParams](#) object to pass to [ccrz.cc\\_hk\\_Payment.createStoredPayment](#).

**Other Inputs Required for ccrz.cc\_hk\_Order.STEP\_CREATE\_STORED\_PAYMENT\_POST**


When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_CREATE_STORED_PAYMENT_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_STORED_PAYMENT_POST,

    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment data
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

[ccrz\\_\\_E\\_Cart\\_\\_c](#) record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the [ccrz\\_\\_E\\_Cart\\_\\_c](#) record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the [ccrz\\_\\_E\\_Cart\\_\\_c](#) record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

[ccrz\\_\\_E\\_Order\\_\\_c](#) record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<[ccrz\\_\\_E\\_OrderItemPricingTier\\_\\_c](#)> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<[ccrz\\_\\_E\\_OrderTerm\\_\\_c](#)> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed to [ccrz.cc\\_hk\\_Payment.createStoredPayment](#).

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

[ccrz.cc\\_hk\\_Payment.TransactionPaymentParams](#) object that the [ccrz.cc\\_hk\\_Payment.createStoredPayment](#) method returns.

**Other Inputs Required for ccrz.cc\_hk\_Order.STEP\_CREATE\_SUBSCRIPTION\_PRE**

When the [ccrz.cc\\_hk\\_Order.PARAM\\_PLACE\\_STEP](#) input's value is [ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_SUBSCRIPTION\\_PRE](#), the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_SUBSCRIPTION_PRE,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment data
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

[ccrz\\_\\_E\\_Cart\\_\\_c](#) record used for placing the order.



**Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the [ccrz\\_\\_E\\_Cart\\_\\_c](#) custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the [ccrz\\_\\_E\\_Cart\\_\\_c](#) record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the [ccrz\\_\\_E\\_Cart\\_\\_c](#) record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values to pass to `ccrz.cc_hk_Subscriptions.onOrder`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object processed from the original `ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA`.

**Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_SUBSCRIPTION_POST`**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is


`ccrz.cc_hk_Order.STEP_CREATE_SUBSCRIPTION_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_SUBSCRIPTION_POST,

    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment data
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<[ccrz\\_\\_E\\_CartItem\\_\\_c](#)> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the [ccrz\\_\\_E\\_Cart\\_\\_c](#) record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

[ccrz\\_\\_E\\_Order\\_\\_c](#) record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<[ccrz\\_\\_E\\_OrderItemPricingTier\\_\\_c](#)> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<[ccrz\\_\\_E\\_OrderTerm\\_\\_c](#)> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

[ccrz.cc\\_hk\\_Payment.TransactionPaymentParams](#) object, updated with the output of the [ccrz.cc\\_hk\\_Subscriptions.onOrder](#) method.

## Other Inputs Required for **ccrz.cc\_hk\_Order.STEP\_CREATE\_TRANSACTION\_PRE**

When the [ccrz.cc\\_hk\\_Order.PARAM\\_PLACE\\_STEP](#) input's value is


[ccrz.cc\\_hk\\_Order.STEP\\_CREATE\\_TRANSACTION\\_PRE](#), the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_TRANSACTION_PRE,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment data
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

[ccrz\\_\\_E\\_Cart\\_\\_c](#) record used for placing the order.



 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values to pass to `ccrz.cc_hk_Order.createTransaction`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object to pass to `ccrz.cc_hk_Order.createTransaction`.

## Other Inputs Required for `ccrz.cc_hk_Order.STEP_CREATE_TRANSACTION_POST`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is


`ccrz.cc_hk_Order.STEP_CREATE_TRANSACTION_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_CREATE_TRANSACTION_POST,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment
})
```

```
data
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed to `ccrz.cc_hk_Order.createTransaction`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object passed to `ccrz.cc_hk_Order.createTransaction`.

**Other Inputs Required for `ccrz.cc_hk_Order.STEP_PAYMENT_PROCESS_PRE`**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is

`ccrz.cc_hk_Order.STEP_PAYMENT_PROCESS_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_PAYMENT_PROCESS_PRE,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
```

```

    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_PAYMENT_RESULT => payment result,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment
    data
  });

```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.



**Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_PAYMENT\_RESULT**

Map<String, Object> that represents the output from `ccrz.cc_hk_Payment.processPayment`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values to pass to `ccrz.ccPaymentProcessor.process`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object to pass to `ccrz.ccPaymentProcessor.process`.

## Other Inputs Required for `ccrz.cc_hk_Order.STEP_PAYMENT_PROCESS_POST`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is

`ccrz.cc_hk_Order.STEP_PAYMENT_PROCESS_POST`, the `place` method's input map requires the following other inputs:

```

ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_PAYMENT_PROCESS_POST,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,

```


```

    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_PAYMENT_RESULT => payment result,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment
    data
  });

```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_PAYMENT\_RESULT**

Map<String, Object> that represents the output from `ccrz.cc_hk_Payment.processPayment`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed to `ccrz.ccPaymentProcessor.process`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object passed to `ccrz.ccPaymentProcessor.process`.

## Other Inputs Required for `ccrz.cc_hk_Order.STEP_UPDATE_ORDER_STATUS_PRE`


When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is

`ccrz.cc_hk_Order.STEP_UPDATE_ORDER_STATUS_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_UPDATE_ORDER_STATUS_PRE,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_ORDER_W_STATUS => order to submit
    ccrz.cc_hk_Order.PARAM_PAYMENT_RESULT => payment result,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment data
});
```

### `ccrz.cc_hk_Order.PARAM_CART`

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

### `ccrz.cc_hk_Order.PARAM_CART_ID`

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_CART_ITEMS`

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

### `ccrz.cc_hk_Order.PARAM_ENC_CART_ID`

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

### `ccrz.cc_hk_Order.PARAM_ORDER`

`ccrz__E_Order__c` record that was inserted.

### `ccrz.cc_hk_Order.PARAM_ORDER_ITEMS`

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

### `ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS`

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

### `ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS`


List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

### `ccrz.cc_hk_Order.PARAM_ORDER_TERMS`

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_W\_STATUS**

`ccrz__E_Order__c` record where the `ccrz__OrderStatus__c` field is ready to update. By default, this step sets the new status to `Order Submitted`.

 **Note:** This method queries the record by only its Salesforce ID.

**ccrz.cc\_hk\_Order.PARAM\_PAYMENT\_RESULT**

`Map<String, Object>` that represents the output from `ccrz.cc_hk_Payment.processPayment`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed to `ccrz.ccPaymentProcessor.process`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object passed to `ccrz.ccPaymentProcessor.process`.

**Other Inputs Required for `ccrz.cc_hk_Order.STEP_UPDATE_ORDER_STATUS_POST`**

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is


`ccrz.cc_hk_Order.STEP_UPDATE_ORDER_STATUS_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_UPDATE_ORDER_STATUS_POST,

    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_ORDER_W_STATUS => order to submit
    ccrz.cc_hk_Order.PARAM_PAYMENT_RESULT => payment result,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment data
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

`List<ccrz__E_CartItem__c>` of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_W\_STATUS**

`ccrz__E_Order__c` record where the `ccrz__OrderStatus__c` field updated to `Order Submitted` or another status that you specify when you override this method in subscriber code.



**Note:** This method queries the record by only its Salesforce ID.

**ccrz.cc\_hk\_Order.PARAM\_PAYMENT\_RESULT**

Map<String, Object> that represents the output from `ccrz.cc_hk_Payment.processPayment`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed to `ccrz.ccPaymentProcessor.process`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object passed to `ccrz.ccPaymentProcessor.process`.

## Other Inputs Required for `ccrz.cc_hk_Order.STEP_INVOICE_CREATE_PRE`


When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is

`ccrz.cc_hk_Order.STEP_INVOICE_CREATE_PRE`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_INVOICE_CREATE_PRE,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_ORDER_W_STATUS => order to submit
    ccrz.cc_hk_Order.PARAM_PAYMENT_RESULT => payment result,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment data
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_W\_STATUS**

`ccrz__E_Order__c` record where the `ccrz__OrderStatus__c` field updated to `Order Submitted` or another status that you specify when you override this method in subscriber code.

 **Note:** This method queries the record by only its Salesforce ID.

**ccrz.cc\_hk\_Order.PARAM\_PAYMENT\_RESULT**

Map<String, Object> that represents the output from `ccrz.cc_hk_Payment.processPayment`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed to `ccrz.ccPaymentProcessor.process`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object passed to `ccrz.ccPaymentProcessor.process`.

## Other Inputs Required for `ccrz.cc_hk_Order.STEP_INVOICE_CREATE_POST`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is

`ccrz.cc_hk_Order.STEP_INVOICE_CREATE_POST`, the `place` method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_INVOICE_CREATE_POST,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
```




```

ccrz.cc_hk_Order.PARAM_INVOICE => invoice,
ccrz.cc_hk_Order.PARAM_INVOICE_ITEMS => list of invoice items
ccrz.cc_hk_Order.PARAM_ORDER => order,
ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
ccrz.cc_hk_Order.PARAM_ORDER_W_STATUS => order to submit
ccrz.cc_hk_Order.PARAM_PAYMENT_RESULT => payment result,
ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment
data
});

```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_INVOICE**

`ccrz__E_Invoice__c` record that the `ccrz.cc_hk_Order.onOrder` method returns in its `ccrz.cc_hk_Order.PARAM_INVOICE` output.

**ccrz.cc\_hk\_Order.PARAM\_INVOICE\_ITEMS**

List<`ccrz__E_InvoiceItem__c`> that the `ccrz.cc_hk_Order.onOrder` method returns in its `ccrz.cc_hk_Order.PARAM_INVOICE_ITEMS` output.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

`ccrz__E_Order__c` record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<`ccrz__E_OrderItem__c`> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<`ccrz__E_OrderItemPricingTier__c`> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<`ccrz__E_OrderTerm__c`> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_W\_STATUS**

`ccrz__E_Order__c` record where the `ccrz__OrderStatus__c` field updated to `Order Submitted` or another status that you specify when you override this method in subscriber code.

 **Note:** This method queries the record by only its Salesforce ID.

**ccrz.cc\_hk\_Order.PARAM\_PAYMENT\_RESULT**

Map<String, Object> that represents the output from `ccrz.cc_hk_Payment.processPayment`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed to `ccrz.ccPaymentProcessor.process`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object passed to `ccrz.ccPaymentProcessor.process`.


## Other Inputs Required for `ccrz.cc_hk_Order.STEP_END`

When the `ccrz.cc_hk_Order.PARAM_PLACE_STEP` input's value is `ccrz.cc_hk_Order.STEP_END`, the place method's input map requires the following other inputs:

```
ccrz.cc_hk_Order.place(new Map<String, Object> {
    ccrz.cc_hk_Order.PARAM_PLACE_STEP => ccrz.cc_hk_Order.STEP_END,
    ccrz.cc_hk_Order.PARAM_CART => cart,
    ccrz.cc_hk_Order.PARAM_CART_ID => cart ID,
    ccrz.cc_hk_Order.PARAM_CART_ITEMS => list of major cart items,
    ccrz.cc_hk_Order.PARAM_ENC_CART_ID => encrypted cart ID,
    ccrz.cc_hk_Order.PARAM_INVOICE => invoice,
    ccrz.cc_hk_Order.PARAM_INVOICE_ITEMS => list of invoice items
    ccrz.cc_hk_Order.PARAM_ORDER => order,
    ccrz.cc_hk_Order.PARAM_ORDER_CHILD_ITEMS => list of minor order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEMS => list of major order items,
    ccrz.cc_hk_Order.PARAM_ORDER_ITEM_TIERS => list of order item pricing tiers,
    ccrz.cc_hk_Order.PARAM_ORDER_TERMS => list of order terms,
    ccrz.cc_hk_Order.PARAM_ORDER_W_STATUS => order to submit
    ccrz.cc_hk_Order.PARAM_PAYMENT_RESULT => payment result,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_DATA => JSON payment data,
    ccrz.cc_hk_Order.PARAM_TRANSACTION_PROCESSED_DATA => serialized transaction payment data
});
```

**ccrz.cc\_hk\_Order.PARAM\_CART**

`ccrz__E_Cart__c` record used for placing the order.

 **Note:** This method evaluates only fields that B2B Commerce for Visualforce defines on the `ccrz__E_Cart__c` custom object. If you've added any custom subscriber fields, query those fields separately.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_CART\_ITEMS**

List<`ccrz__E_CartItem__c`> of the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the `ccrz__E_Cart__c` record used for placing the order.

**ccrz.cc\_hk\_Order.PARAM\_INVOICE**

`ccrz__E_Invoice__c` record that the `ccrz.cc_hk_Invoice.onOrder` method returns in its `ccrz.cc_hk_Invoice.PARAM_INVOICE` output.

**ccrz.cc\_hk\_Order.PARAM\_INVOICE\_ITEMS**

List<[ccrz\\_\\_E\\_InvoiceItem\\_\\_c](#)> that the `ccrz.cc_hk_Invoice.onOrder` method returns in its `ccrz.cc_hk_Invoice.PARAM_INVOICE_ITEMS` output.

**ccrz.cc\_hk\_Order.PARAM\_ORDER**

[ccrz\\_\\_E\\_Order\\_\\_c](#) record that was inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEMS**

List<[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)> of the order items that were inserted and that correspond to the cart's major line items, such as standard products, composite products, or coupons.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_CHILD\_ITEMS**

List<[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)> of the order's minor line items that were inserted and that correspond to the cart's minor line items, such as component products.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ITEM\_TIERS**

List<[ccrz\\_\\_E\\_OrderItemPricingTier\\_\\_c](#)> of the order items' pricing tiers that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_TERMS**

List<[ccrz\\_\\_E\\_OrderTerm\\_\\_c](#)> of the order terms that were inserted.

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_W\_STATUS**

[ccrz\\_\\_E\\_Order\\_\\_c](#) record where the `ccrz__OrderStatus__c` field updated to `Order Submitted` or another status that you specify when you override this method in subscriber code.



**Note:** This method queries the record by only its Salesforce ID.

**ccrz.cc\_hk\_Order.PARAM\_PAYMENT\_RESULT**

Map<[String](#), [Object](#)> that represents the output from `ccrz.cc_hk_Payment.processPayment`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed to `ccrz.ccPaymentProcessor.process`.

**ccrz.cc\_hk\_Order.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object passed to `ccrz.ccPaymentProcessor.process`.

## Outputs

Map<[String](#), [Object](#)> that doesn't include any new output data.

### SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz\\_\\_E\\_CartItem\\_\\_c](#)

[ccrz\\_\\_E\\_Coupon\\_\\_c](#)

[ccrz\\_\\_E\\_Order\\_\\_c](#)

[ccrz\\_\\_E\\_OrderItem\\_\\_c](#)

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

`ccrz.ccApiCart.place`

`ccrz.cc_hk_Payment`

`ccrz.ccPaymentProcessor`

[Salesforce Help](#) Generate an Invoice for a Buyer's Order

## ccrz.cc\_hk\_Order.placeTarget

Appends an order query string parameter to the URL for the storefront's Order Confirmation page. To fully customize a buyer's navigation path when they complete checkout, override this method in your extension subscriber code.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global virtual Map<String, Object> placeTarget (Map<String, Object>)
```

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.cc\_hk\_Payment.PARAM\_ORDER**

`ccrz__E_Order__c` record for the order that the buyer placed.

 **Important:** Ensure that this record has the `ccrz__EncryptedId__c` field set.

### Logic Performed

This method completes the following actions:

1. Returns the value of the Order Confirmation URL configuration of the Checkout module (`co.cnfurl`) in your storefront configuration settings.
2. Appends the query string parameter `?o=` or `&o=`, and adds the order's encrypted ID.

For example, let's say that the Order Confirmation URL configuration specifies the following custom URL instead of the storefront's default Order Confirmation page:

```
https://orders.capricorncoffee.us/processOrder
```

This method returns the full URL, which includes the query string parameter:

```
https://orders.capricorncoffee.us/processOrder?o=orderEncryptedId
```

### Outputs

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_hk\_Order.PARAM\_ORDER\_ID**

String that specifies the encrypted ID of the order.

**ccrz.cc\_hk\_Order.PARAM\_PLACE\_TARGET**

The full URL to the Order Confirmation page, including the order query string parameter.

SEE ALSO:

[Salesforce Help Parameters in Storefront Page URLs](#)

## ccrz.cc\_hk\_Order.reorder

Defines legacy behavior for copying the data for a completed order and its order items to a new cart and cart items.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Important:

- This method defines legacy behavior (by default, the managed package instead references `ccrz.ccApiOrder.reorder`). To reference this legacy method in your solution, extend the default `ccrz.ccLogicOrderReOrder` class so that it calls this method. The managed package provides an example extension, `ccrz.ccLogicOrderReOrder7HK`, that you can use as a starting point for extending `ccrz.ccLogicOrderReOrder`.
- When you use `ccrz.ccLogicOrderReOrder7HK` or your own extension, the My Orders section of the My Account page calls this method when a buyer clicks the Reorder button.

### Signature

```
global virtual Map<String, Object> reorder (Map<String, Object>)
```

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.cc\_hk\_Payment.PARAM\_ORDER\_ID**

String that specifies the 18-digit Salesforce ID of the order.

## Outputs

Map<[String](#), [Object](#)> that can include the following keys:

### **ccrz.cc\_hk\_Order.PARAM\_ENC\_CART\_ID**

String that specifies the encrypted ID of the created [ccrz\\_\\_E\\_Cart\\_\\_c](#) record.

SEE ALSO:

[ccrz.ccApiOrder.reorder](#)

[ccrz.ccLogicOrderReOrder](#)

[ccrz.ccLogicOrderReOrder7HK Example Extension Class](#)

[Salesforce Help Allow a Buyer to Manage Their Orders on the My Account Page](#)

## ccrz.cc\_hk\_Payment

Evaluate and maintain stored payments and transaction payments.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## TransactionPaymentParams Inner Class

The [ccrz.cc\\_hk\\_Payment](#) extension point class defines an inner class, [TransactionPaymentParams](#), that defines the following Apex properties. For more information about Apex properties and accessors, see [Apex Properties](#).

### **accountNumber {get; set;}**

String that specifies an account number for the payment.

- For a purchase order (PO), this value is the entire PO number.
- For a credit card or bank account, this value is likely the account number's last four digits.

### **accountType {get; set;}**

String that specifies the method of payment, such as purchase order (PO), credit card, and so on.

### **amount {get; set;}**

Decimal that specifies the transaction's actual payment amount.

### **comments {get; set;}**

String that specifies free text associated with the payment.

**displayName {get; set;}**

String that specifies the user-entered name for the stored payment that saves this transaction payment information.

**expirationMonth {get; set;}**

String that specifies the two-digit expiration month for the payment, such as for a credit card expiration date.

**expirationYear {get; set;}**

String that specifies the four-digit expiration year for the payment, such as for a credit card expiration date.

**paymentType {get; set;}**

String that specifies more specific payment method details for `accountType`. For example, if `accountType` is `Credit Card`, then this value can be the type of credit card.

**requestAmount {get; set;}**

Decimal that specifies a requested payment amount.

**saveAsStoredPayment {get; set;}**

Boolean

Value	Usage
<i>true</i>	Save the transaction payment information as a stored payment for the account.
<i>false</i> (default)	Don't save the transaction payment information as a stored payment.

**storedPaymentId {get; set;}**

String that specifies the Salesforce ID of the original `ccrz__E_StoredPayment__c` for the payment, if applicable.

**storefront {get; set;}**

String that specifies the storefront for the payment.

**subAccountNumber {get; set;}**

String that specifies a subaccount used with `accountNumber`.

**token {get; set;}**

String that specifies the tokenized form of the payment information.

**transactionCode {get; set;}**

String that specifies a code associated with the transaction. For example, for a credit card payment, this value is the specific transaction ID.

**transactionPaymentId {get; set;}**

String that specifies a unique ID that associates the record with an external system.

**transactionSubcode {get; set;}**

String that specifies a secondary code associated with the transaction. For example, if the transaction failed, this value can be failure's error code.

**transactionType {get; set;}**

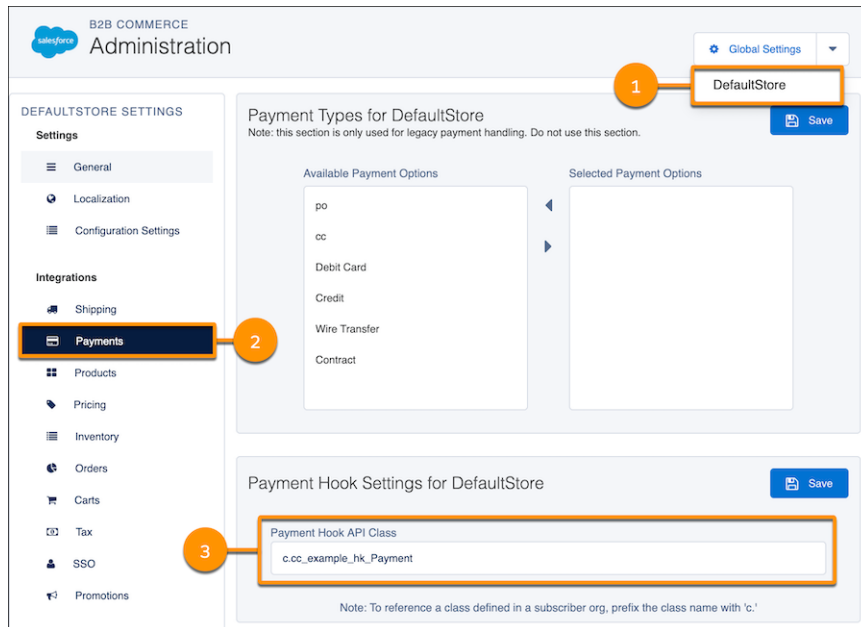
String that specifies a provider-specific type of transaction, such as `AUTH` or `CAPTURE` for a credit card payment.

**verificationCode {get; set;}**

String that specifies a secondary `transactionCode`.

## Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Payments** in the Integrations section of the Storefront Settings menu (2). Update the Payment Hook API Class field (3) with your custom class name, prefixed with `c.`, such as `c.cc_example_hk_Payment`.



**Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

**Note:** The Payment Types for *Storefront* section of the Payments integrations page is deprecated.

## Methods

### IN THIS SECTION:

#### [convertStoredPaymentToMap](#)

Parses data on a specified `ccrz__E_StoredPayment__c` record and returns a `Map<String, Object>`.

#### [convertTxnPaymentToMap](#)

Parses data on a specified `ccrz__E_TransactionPayment__c` record and returns a `Map<String, Object>`.

#### [convertTxnPaymentRefToMap](#)

Parses only the reference data on a specified `ccrz__E_TransactionPayment__c` record and returns a `Map<String, Object>`.

#### [createStoredPayment](#)

Creates a stored payment from transaction payment details during checkout. This method is also called when a buyer creates a purchase order (PO) from the My Wallet section of the My Account page. For custom payment implementations not defined in the managed package, this method isn't called. This method does *not* support Apex callouts.



[deserializeTransactionParams](#)

Parses JSON input and return a `ccrz.cc_hk_Payment.TransactionPaymentParams` object.

[fetchPaymentTypes](#)

Returns all configuration settings defined in each payment type's respective configuration module.

[fetchStoredPayments](#)

Returns the stored payments available for checkout, invoices, or the My Wallet section of the My Account page.

[fetchTransactionPayment](#)

Provides an extension point for showing transaction payment data on the storefront's Transaction Payment Detail page.

[getApplicablePaymentTypes](#)

Provides an extension point for overriding the allowed payment types for checkout, invoices, or the My Wallet section of the My Account page.

[getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Payment` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

[getPaymentProcessor](#)

Creates an instance of the `ccrz.ccPaymentProcessor` class for the transaction's `accountType`.

[processPayment](#)

Prepares transaction payment data during checkout, and passes that payment information to the `ccrz.cc_hk_Order.createTransaction` method. This method also provides an extension point for implementing validation of a custom payment type. This method supports Apex callouts.

[removeStoredPayment](#)

Deletes a stored payment from the My Wallet section of the My Account page.

## SEE ALSO:

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

## ccrz.cc\_hk\_Payment.convertStoredPaymentToMap

Parses data on a specified `ccrz__E_StoredPayment__c` record and returns a `Map<String, Object>`.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> convertStoredPaymentToMap(ccrz__E_StoredPayment__c)
```

## Inputs (Required)

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#) record to evaluate.

## Outputs

Map<String, Object> that includes the following keys, which correspond to fields on the input [ccrz\\_\\_E\\_StoredPayment\\_\\_c](#) record:

- '**accountNumber**'  
String that corresponds to the [ccrz\\_\\_AccountNumber\\_\\_c](#) field.
- '**accountType**'  
String that corresponds to the [ccrz\\_\\_AccountType\\_\\_c](#) field.
- '**displayName**'  
String that corresponds to the [ccrz\\_\\_DisplayName\\_\\_c](#) field.
- '**expirationMonth**'  
Integer that corresponds to the [ccrz\\_\\_ExpMonth\\_\\_c](#) field.
- '**expirationYear**'  
Integer that corresponds to the [ccrz\\_\\_ExpYear\\_\\_c](#) field.
- '**isDefault**'  
Boolean that corresponds to the [ccrz\\_\\_Default\\_\\_c](#) field.
- '**isEnabled**'  
Boolean that corresponds to the [ccrz\\_\\_Enabled\\_\\_c](#) field.
- '**name**'  
String that corresponds to the Name field.
- '**paymentType**'  
String that corresponds to the [ccrz\\_\\_PaymentType\\_\\_c](#) field.
- '**readOnly**'  
Boolean that corresponds to the [ccrz\\_\\_ReadOnly\\_\\_c](#) field.
- '**sequence**'  
Integer that corresponds to the [ccrz\\_\\_Sequence\\_\\_c](#) field.
- '**sfid**'  
String that corresponds to the Salesforce ID.
- '**subaccountNumber**'  
String that corresponds to the [ccrz\\_\\_SubaccountNumber\\_\\_c](#) field.
- '**token**'  
String that corresponds to the [ccrz\\_\\_Token\\_\\_c](#) field.

SEE ALSO:

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

## ccrz.cc\_hk\_Payment.convertTxnPaymentToMap

Parses data on a specified `ccrz__E_TransactionPayment__c` record and returns a `Map<String, Object>`.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global static Map<String, Object> convertTxnPaymentToMap(ccrz__E_TransactionPayment__c)
```

### Inputs (Required)

`ccrz__E_TransactionPayment__c` record to evaluate.

### Outputs

`Map<String, Object>` that includes the following keys, which correspond to fields on the input `ccrz__E_TransactionPayment__c` record:

- '**account**'  
ID that corresponds to the `ccrz__Account__c` field.
- '**accountNumber**'  
String that corresponds to the `ccrz__AccountNumber__c` field.
- '**accountType**'  
String that corresponds to the `ccrz__AccountType__c` field.
- '**amount**'  
Decimal that corresponds to the `ccrz__Amount__c` field.
- '**billTo**'  
ID that corresponds to the `ccrz__BillTo__c` field.
- '**ccOrder**'  
ID that corresponds to the `ccrz__CCOrder__c` field.
- '**currencyIsoCode**'  
String that corresponds to the `ccrz__CurrencyISOCODE__c` field.
- '**expirationMonth**'  
Integer that corresponds to the `ccrz__ExpirationMonth__c` field.

- '**expirationYear**'  
Integer that corresponds to the `ccrz__ExpirationYear__c` field.
- '**name**'  
String that corresponds to the Name field.
- '**invoiceAppliedFor**'  
ID that corresponds to the `ccrz__InvoiceAppliedFor__c` field.
- '**invoiceAppliedTo**'  
ID that corresponds to the `ccrz__InvoiceAppliedTo__c` field.
- '**parent**'  
ID that corresponds to the `ccrz__ParentTransactionPayment__c` field.
- '**paymentTS**'  
Datetime that corresponds to the `ccrz__TransactionTS__c` field.
- '**paymentTSStr**'  
String that corresponds to the `ccrz__TransactionTS__c` field.
- '**paymentType**'  
String that corresponds to the `ccrz__PaymentType__c` field.
- '**requestAmount**'  
Decimal that corresponds to the `ccrz__RequestAmount__c` field.
- '**sfid**'  
String that corresponds to the Salesforce ID.
- '**soldTo**'  
ID that corresponds to the `ccrz__SoldTo__c` field.
- '**sourceTransactionPayment**'  
ID that corresponds to the `ccrz__SourceTransactionPayment__c` field.
- '**subAccountNumber**'  
String that corresponds to the `ccrz__SubAccountNumber__c` field.
- '**transactionPaymentId**'  
String that corresponds to the `ccrz__TransactionPaymentId__c` field.
- '**transactionType**'  
String that corresponds to the `ccrz__TransactionType__c` field.

SEE ALSO:

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

[Salesforce Help](#) Generate an Invoice for a Buyer's Order

## ccrz.cc\_hk\_Payment.convertTxnPaymentRefToMap

Parses only the reference data on a specified `ccrz__E_TransactionPayment__c` record and returns a `Map<String, Object>`.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> convertTxnPaymentRefToMap(ccrz__E_TransactionPayment__c)
```

## Inputs (Required)

`ccrz__E_TransactionPayment__c` record to evaluate.

## Outputs

Map<String, Object> that includes the following keys, which correspond to fields on the input `ccrz__E_TransactionPayment__c` record:

**'name'**

String that corresponds to the Name field.

**'sfid'**

String that corresponds to the Salesforce ID.

**'transactionPaymentId'**

String that corresponds to the `ccrz__TransactionPaymentId__c` field.

SEE ALSO:

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

[Salesforce Help Generate an Invoice for a Buyer's Order](#)

## ccrz.cc\_hk\_Payment.createStoredPayment

Creates a stored payment from transaction payment details during checkout. This method is also called when a buyer creates a purchase order (PO) from the My Wallet section of the My Account page. For custom payment implementations not defined in the managed package, this method isn't called. This method does *not* support Apex callouts.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> createStoredPayment (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.cc\_hk\_Payment.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object that contains the transaction payment details.

## Outputs

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_hk\_Payment.PARAM\_TRANSACTION\_PROCESSED\_DATA**

An instance of `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA` that contains the input transaction payment details.

This method also creates a `ccrz__E_StoredPayment__c` record, sets the following fields, and executes the DML `insert` operation on the record:

Field	Source Value
<code>ccrz__Account__c</code>	<code>ccrz.cc_CallContext.currAccountId</code>
<code>ccrz__AccountNumber__c</code>	accountNumber field of <code>ccrz.cc_hk_Payment.TransactionPaymentParams</code>
<code>ccrz__AccountType__c</code>	accountType field of <code>ccrz.cc_hk_Payment.TransactionPaymentParams</code>
<code>ccrz__DisplayName__c</code>	displayName field of <code>ccrz.cc_hk_Payment.TransactionPaymentParams</code>
<code>ccrz__EffectiveAccountID__c</code>	<code>ccrz.cc_CallContext.effAccountId</code>
<code>ccrz__Enabled__c</code>	<code>true</code>
<code>ccrz__ExpMonth__c</code>	expirationMonth field of <code>ccrz.cc_hk_Payment.TransactionPaymentParams</code>
<code>ccrz__ExpYear__c</code>	expirationYear field of <code>ccrz.cc_hk_Payment.TransactionPaymentParams</code>

Field	Source Value
ccrz__PaymentType__c	paymentType field of ccrz.cc_hk_Payment.TransactionPaymentParams
ccrz__Storefront__c	<a href="#">ccrz.cc_CallContext.storefront</a>
ccrz__SubaccountNumber__c	subAccountNumber field of ccrz.cc_hk_Payment.TransactionPaymentParams
ccrz__Token__c	subAccountNumber field of ccrz.cc_hk_Payment.TransactionPaymentParams
ccrz__User__c	<a href="#">ccrz.cc_CallContext.currUserId</a>
OwnerId	<a href="#">ccrz.cc_CallContext.currUserId</a>

## SEE ALSO:

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

[Inserting and Updating Records](#)

[Salesforce Help Configure Payment Options Available on the Checkout Page](#)

[Salesforce Help Allow a Buyer to Manage Stored Payments on the My Account Page](#)

## ccrz.cc\_hk\_Payment.deserializeTransactionParams

Parses JSON input and return a `ccrz.cc_hk_Payment.TransactionPaymentParams` object.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global static ccrz.cc_hk_Payment.TransactionPaymentParams
deserializeTransactionParams (String)
```

## Inputs (Required)

### **transactionParams**

JSON-formatted string of transaction payment properties and their values.

## Outputs

`ccrz.cc_hk_Payment.TransactionPaymentParams` object.

SEE ALSO:

[ccrz.cc\\_hk\\_Payment](#)

## ccrz.cc\_hk\_Payment.fetchPaymentTypes

Returns all configuration settings defined in each payment type's respective configuration module.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchPaymentTypes (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### **ccrz.cc\_hk\_Payment.PARAM\_APPL\_PAYMENT\_TYPES**

String that specifies a comma-separated list of payment types, such as *po, credit, wire*.

## Outputs

`Map<String, Object>` that can include the following keys:

### **ccrz.cc\_hk\_Payment.PARAM\_PAYMENT\_TYPES**

`Map<String, Map<String, Object>>` where:

- Each key in the outer map corresponds to one payment type specified in `ccrz.cc_hk_Payment.PARAM_APPL_PAYMENT_TYPES`, such as *po*.



- Each inner map describes a key-value pair of a payment type's configuration and its value.

## SEE ALSO:

- [Salesforce Help Manage Configuration Settings](#)
- [Salesforce Help Set Up Payment Types for Storefront Purchases](#)
- [Salesforce Help Configure Payment Options Available on the Checkout Page](#)
- [Salesforce Help Allow a Buyer to Manage Stored Payments on the My Account Page](#)
- [Salesforce Help Generate an Invoice for a Buyer's Order](#)

## ccrz.cc\_hk\_Payment.fetchStoredPayments

Returns the stored payments available for checkout, invoices, or the My Wallet section of the My Account page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Important:** In B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12), this method defines legacy behavior. The managed package instead references the `ccrz.ccApiStoredPayment.fetch` global API method by default. To continue using existing extensions of this method, change the Use Payment Hook configuration of the My Wallet module in your storefront configuration settings to `TRUE`. For new solutions, extend the `ccrz.ccServiceStoredPayment` data service provider class.

### Signature

```
global virtual Map<String, Object> fetchStoredPayments (Map<String, Object>)
```

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

#### `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX`

String that specifies the one of the following values for the context of the payment:

- `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_CHECKOUT` for a payment made during checkout. By default, this key's value is `co`.

- `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_INVOICE` for a payment made against an invoice. By default, this key's value is `inv`.
- `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_WALLET` for a stored payment in the My Wallet section of the My Account page. By default, this key's value is `w1t`.

## Logic Performed

This method completes the following actions:

1. Calls the `ccrz.cc_hk_Payment.getApplicablePaymentTypes` method for returning the allowed payment types for the specified `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX`.
2. Queries for `ccrz__E_StoredPayment__c` records that exist for those allowed types.
3. Calls the `ccrz.cc_hk_Payment.convertStoredPaymentToMap` method for parsing each `ccrz__E_StoredPayment__c` record and returning its data as a `Map<String, Object>`.

## Outputs

`Map<String, Object>` that can include the following keys:

### **ccrz.cc\_hk\_Payment.PARAM\_STORED\_PAYMENTS**

`List<Map<String, Object>>` where each `Map<String, Object>` represents a `ccrz__E_StoredPayment__c` record, which can be serialized to JSON.

SEE ALSO:

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

[ccrz.cc\\_hk\\_Payment.getApplicablePaymentTypes](#)

[ccrz.cc\\_hk\\_Payment.convertStoredPaymentToMap](#)

[Salesforce Help Configure Payment Options Available on the Checkout Page](#)

[Salesforce Help Allow a Buyer to Manage Stored Payments on the My Account Page](#)

## ccrz.cc\_hk\_Payment.fetchTransactionPayment

Provides an extension point for showing transaction payment data on the storefront's Transaction Payment Detail page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> fetchTransactionPayment (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.cc\_hk\_Payment.PARAM\_PAYMENTID**  
`ccrz__E_TransactionPayment__c` record to evaluate.

This method passes the `ccrz__E_TransactionPayment__c` to the `ccrz.cc_hk_Payment.convertTxnPaymentToMap` method, which parses the record and returns a `Map<String, Object>` of its data.

## Outputs

`Map<String, Object>` that can include the following keys:

**ccrz.cc\_hk\_Payment.PARAM\_PAYMENT**  
`Map<String, Object>` that contains the results of the `ccrz.cc_hk_Payment.convertTxnPaymentToMap` method. By default, this return data includes:

- The transaction payment.
- Any child transaction payments, which can exist when a single transaction applies to multiple records. For example, when similar transaction details apply to multiple payments, a parent transaction can contain the shared details, but child transactions can specify the individual payments.
- Any source transaction payments, which can exist for a chain of transactions. For example, in an authorization and capture chain, the authorization is the source transaction payment and the capture is the current transaction payment.

SEE ALSO:

[ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#)

## ccrz.cc\_hk\_Payment.getApplicablePaymentTypes

Provides an extension point for overriding the allowed payment types for checkout, invoices, or the My Wallet section of the My Account page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> getApplicablePaymentTypes (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### **ccrz.cc\_hk\_Payment.PARAM\_APPL\_PAYMENT\_TYPES**

String that specifies a comma-separated list of allowed payment types for the specified `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX`, such as *po, credit, wire*.

### **ccrz.cc\_hk\_Payment.PARAM\_PAYMENT\_CTX**

String that specifies the one of the following values for the context of the payment:

- `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_CHECKOUT` for a payment made during checkout. By default, this key's value is `co`, which matches the API name for the [Checkout configuration module](#).
- `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_INVOICE` for a payment made against an invoice. By default, this key's value is `inv`, which matches the API name for the [Invoices configuration module](#).
- `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_WALLET` for a stored payment in the My Wallet section of the My Account page. By default, this key's value is `w1t`, which matches the API name for the [My Wallet configuration module](#).

## Logic Performed

This method looks for a storefront configuration with the name `pmts` in the configuration module that matches the specified context. For example, for checkout, this method looks for the configuration `co.pmts`.

## Outputs

`Map<String, Object>` that can include the following keys:

### **ccrz.cc\_hk\_Payment.PARAM\_APPL\_PAYMENT\_TYPES**

String that specifies a comma-separated list of allowed payment types for the specified `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX`, updated with the result of the method.

### SEE ALSO:

- [Salesforce Help Manage Configuration Settings](#)
- [Salesforce Help Set Up Payment Types for Storefront Purchases](#)
- [Salesforce Help Configure Payment Options Available on the Checkout Page](#)
- [Salesforce Help Allow a Buyer to Manage Stored Payments on the My Account Page](#)
- [Salesforce Help Generate an Invoice for a Buyer's Order](#)

## ccrz.cc\_hk\_Payment.getInstance

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_Payment` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static ccrz.cc_hk_Payment getInstance (Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_Payment.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_hk_Payment`, or your custom extension point class.

## `ccrz.cc_hk_Payment.getPaymentProcessor`

Creates an instance of the `ccrz.ccPaymentProcessor` class for the transaction's `accountType`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> getPaymentProcessor (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.cc_hk_Payment.PARAM_ACCOUNT_TYPE`

String, based on the input data's `accountType`, that specifies the transaction's method of payment, such as `po` for purchase order or `credit` for credit card.

## Logic Performed

This method looks for a storefront configuration, `pmt_key.proc`, where **key** is a configured payment type for the storefront, such as `po` for a purchase order, that matches `ccrz.cc_hk_Payment.PARAM_ACCOUNT_TYPE`. For more information about payment type configurations, see [Set Up Payment Types for Storefront Purchases](#).

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_Payment.PARAM_PAYMENT_PROCESSOR`

`ccrz.ccPaymentProcessor` instance configured for the specified `ccrz.cc_hk_Payment.PARAM_ACCOUNT_TYPE`. For example, if the account type is a purchase order, this method returns the `ccrz.ccPaymentProcessor` for purchase orders.

SEE ALSO:

[ccrz.ccPaymentProcessor](#)

[Salesforce Help Set Up Payment Types for Storefront Purchases](#)

## `ccrz.cc_hk_Payment.processPayment`

Prepares transaction payment data during checkout, and passes that payment information to the `ccrz.cc_hk_Order.createTransaction` method. This method also provides an extension point for implementing validation of a custom payment type. This method supports Apex callouts.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> processPayment (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.cc_hk_Payment.PARAM_CART`

`ccrz__E_Cart__c` record that's being converted to an order.

### `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA`

JSON-formatted string of transaction payment properties and their values, which the `ccrz.cc_hk_Payment.deserializeTransactionParams` method parses into a `ccrz.cc_hk_Payment.TransactionPaymentParams` object.



**Note:** To include custom payment fields during checkout, override this method in your custom subscriber class. Be sure to deserialize the string to a different object.

## Logic Performed

This method completes the following actions:

1. Sets the transaction payment's `amount` based on the cart's `ccrz__TotalAmount__c` value.
2. Evaluates the transaction input data's `accountType` and sets the `ccrz.cc_hk_Payment.PARAM_ACCOUNT_TYPE` key.
3. Invokes the `ccrz.cc_hk_Payment.getPaymentProcessor` method for returning the `ccrz.ccPaymentProcessor` configured for the specified `accountType`.
4. Invokes the `ccrz.ccPaymentProcessor.preProcess` method for handling any Apex callouts for the specified `accountType`. If the transaction's input data specified a `storedPaymentId`, this method sets the `ccrz.cc_hk_Payment.PARAM_STORED_PAYMENT` key in the input map for `ccrz.ccPaymentProcessor.preProcess`.



**Important:** When you [implement a custom payment type](#), be sure to override this method in your extension implementation to include validation for the payment. The exact validation logic required depends on the conventions and requirements of the payment type.

## Outputs

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_Payment.PARAM_ACCOUNT_TYPE`

String, based on the input data's `accountType`, that specifies the transaction's method of payment, such as purchase order (PO), credit card, and so on. This value is passed to the `ccrz.ccPaymentProcessor.preProcess` method.

### `ccrz.cc_hk_Payment.PARAM_PAYMENT_PROCESSOR`

`ccrz.ccPaymentProcessor` instance configured for the specified `accountType`, which the `ccrz.cc_hk_Payment.getPaymentProcessor` method returned.

### `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA`

A copy of the original input `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA`.

**ccrz.cc\_hk\_Payment.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object that represents the original input `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA`, updated with the cart's amount set.

## SEE ALSO:

[ccrz.ccPaymentProcessor](#)

[ccrz.ccPaymentProcessor.preProcess](#)

[ccrz.cc\\_hk\\_Order.createTransaction](#)

[Salesforce Help Set Up Payment Types for Storefront Purchases](#)

[Invoking Callouts Using Apex](#)

**ccrz.cc\_hk\_Payment.removeStoredPayment**

Deletes a stored payment from the My Wallet section of the My Account page.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**!** **Important:** In B2B Commerce for Visualforce Winter '21 (managed package version 4.13 and API version 12), this method defines legacy behavior. The managed package instead references the `ccrz.ccLogicStoredPaymentRemove` logic service provider class by default. To continue using existing extensions of this method, change the Use Payment Hook configuration of the My Wallet module in your storefront configuration settings to `TRUE`. For new solutions, extend the `ccrz.ccLogicStoredPaymentRemove` logic service provider class.

**Signature**

```
global virtual Map<String, Object> removeStoredPayment (Map<String, Object>)
```

**Inputs (Required)**

`Map<String, Object>` that must include the following *required* keys:


**ccrz.cc\_hk\_Payment.PARAM\_STORED\_PAYMENT\_ID**

String that specifies the Salesforce ID of the `ccrz__E_StoredPayment__c` record to delete.



## Outputs

Boolean

Value	Usage
true	The call completed.
false	The call encountered errors.   <b>Tip:</b> B2B Commerce for Visualforce doesn't always return an exception for any errors that can occur. When this value is <code>false</code> , <a href="#">consider rolling back the API transaction to a previous savepoint</a> .

SEE ALSO:

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)


[Salesforce Help Allow a Buyer to Manage Stored Payments on the My Account Page](#)

## ccrz.cc\_hk\_Pricing

Implement subscriber-defined behavior for pricing products, including entitlements, contracts, and other price rules. This class defines a legacy behavior. For new implements, extend the `ccrz.ccLogicProductPricing` logic service provider instead.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.cc\\_hk\\_Pricing](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:


[ccrz.ccLogicProductPricing](#)

## ccrz.cc\_hk\_SSO

Implement subscriber-defined behavior for overriding default storefront URLs, such as to add a single sign-on (SSO) service for logging in.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.cc\\_hk\\_SSO](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## ccrz.cc\_hk\_Subscriptions

Implement subscriber-defined behavior for overriding the default business logic for processing subscriptions and creating installment orders.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.cc\\_hk\\_Subscriptions](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## ccrz.cc\_hk\_TaxCalculation

Calculate the tax amount on a particular cart from the Order Review section of the storefront's Checkout page. The calculated tax is based on the taxable products included in the cart and the cart's shipping address.

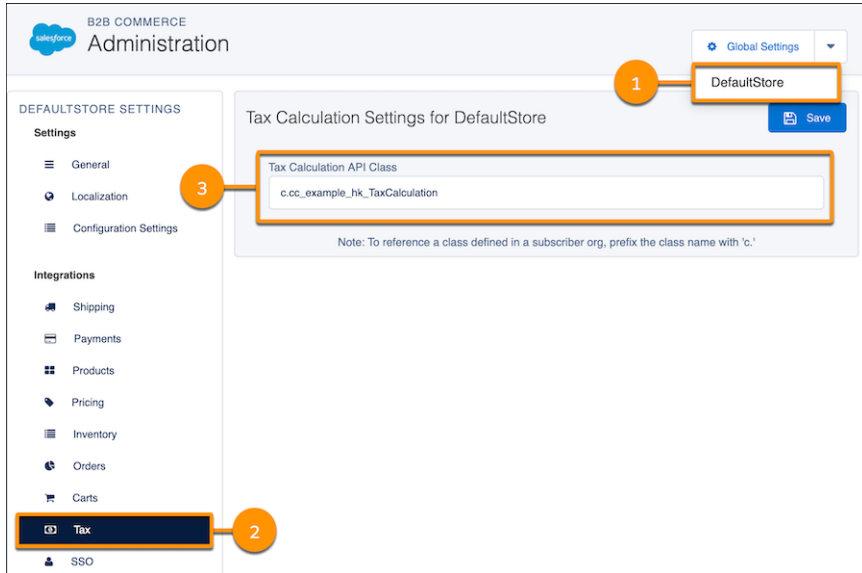
### Compatibility


This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Extension Class Setting in CC Admin

If you extend the default version of this class, update your storefront settings in CC Admin. Select your storefront (1), and then select **Tax** in the Integrations section of the Storefront Settings menu (2). Update the Tax Calculation API Class field (3) with your custom class name, prefixed with *c.*, such as *c.cc\_example\_hk\_TaxCalculation*.



 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

## Methods

### IN THIS SECTION:

#### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_TaxCalculation` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

#### [getTaxAmount](#)

Returns the tax amount for the current cart. This method only executes when the `ccrz.cc_hk_skipTaxCalculation` method returns `false`. To integrate an external tax service, override this method in your subscriber code.

#### [skipTaxCalculation](#)

Determines whether the Checkout page calculates tax for the current cart.

#### [Avalara AvaTax SDK Example for ccrz.cc\\_hk\\_TaxCalculation](#)

To implement the Avalara AvaTax software development kit (SDK) for storefront tax calculation, create custom subscriber code that extends the default `ccrz.cc_hk_TaxCalculation` class. The following code shows examples of overriding the `getTaxAmount` method.

### SEE ALSO:

[Salesforce Help Show Fees and Charges in the Order Review Section of the Checkout Page](#)

## `ccrz.cc_hk_TaxCalculation.getInstance`

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_TaxCalculation` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static ccrz.cc_hk_TaxCalculation getInstance(Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### **ccrz.cc\_hk\_TaxCalculation.PARAM\_STOREFRONT**

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_hk_TaxCalculation`, or your custom extension point class.

## ccrz.cc\_hk\_TaxCalculation.getTaxAmount

Returns the tax amount for the current cart. This method only executes when the `ccrz.cc_hk_skipTaxCalculation` method returns `false`. To integrate an external tax service, override this method in your subscriber code.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Decimal getTaxAmount (Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.cc_hk_TaxCalculation.CART_ID`

String that specifies the 18-digit Salesforce ID of the cart to evaluate.

### `ccrz.cc_hk_TaxCalculation.PARAM_SHIPADDR`

`ccrz.cc_bean_MockContactAddress` that

## Outputs

Decimal value of the calculated tax amount.



Example:

SEE ALSO:

[ccrz\\_\\_E\\_TaxRate\\_\\_c](#)

## `ccrz.cc_hk_TaxCalculation.skipTaxCalculation`

Determines whether the Checkout page calculates tax for the current cart.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Boolean skipTaxCalculation (Map<String, Object>)
```

## Inputs (Optional)

None

## Outputs

Boolean

Value	Usage
true	<p>Either of the following conditions are true:</p> <ul style="list-style-type: none"> <li>The Skip Tax Calculation option in the storefront general settings is enabled.</li> <li>The current account has the <code>ccrz__TaxExemptAccount__c</code> field enabled.</li> </ul> <p>The <code>ccrz.cc_hk_TaxCalculation.getTaxAmount</code> method isn't called.</p>
false	<p>Both of the following conditions are true:</p> <ul style="list-style-type: none"> <li>The Skip Tax Calculation option in the storefront general settings is disabled.</li> <li>The current account has the <code>ccrz__TaxExemptAccount__c</code> field disabled.</li> </ul> <p>The <code>ccrz.cc_hk_TaxCalculation.getTaxAmount</code> method is called for returning the current cart's applicable tax amount.</p>

SEE ALSO:


- [Account \(B2B Commerce for Visualforce\)](#)
- [Salesforce Help Storefront General Settings](#)

## Avalara AvaTax SDK Example for ccrz.cc\_hk\_TaxCalculation

To implement the Avalara AvaTax software development kit (SDK) for storefront tax calculation, create custom subscriber code that extends the default `ccrz.cc_hk_TaxCalculation` class. The following code shows examples of overriding the `getTaxAmount` method.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [Sample Code using Avalara AvaTax SDK](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

- [ccrz.cc\\_hk\\_TaxCalculation.getTaxAmount](#)
- [Salesforce Help Market Templates for B2B Commerce for Visualforce](#)

## ccrz.cc\_hk\_UserInterface

Specify the JavaScript and CSS libraries referenced in each storefront page, including standard includes, proprietary libraries, and theme-specific files in a static resource.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## ∇004 Inner Class

The `ccrz.cc_hk_UserInterface` extension point class defines an inner class, `∇004`, that specifies the most recent default libraries and logic for evaluating configuration setting values. The `∇004` inner class extends the `∇003` and `∇002` inner classes, which are deprecated.

 **Note:** The reference for each method specifies the behavior for only the `∇004` inner class.


## Extension Class Setting in CC Admin

If you extend the default version of this class, extend the `∇004` inner class:

```
global virtual with sharing class cc_example_hk_UserInterface extends
ccrz.cc_hk_UserInterface.∇004 {
    global cc_example_hk_UserInterface {
        // Your custom logic
    }
}
```

Then, update your storefront settings in CC Admin. Select your storefront (1), and then select **Themes** in the Integrations section of the Storefront Settings menu (2). Update the User Interface Extension API Class field (3) with your custom class name, prefixed with `c.`, such as `c.cc_example_hk_UserInterface`.

The screenshot shows the B2B Commerce Administration interface. The top navigation bar includes the Salesforce logo, 'B2B COMMERCE Administration', and a 'Global Settings' dropdown menu. A callout '1' points to the 'DefaultStore' dropdown menu. The left sidebar contains a 'DEFAULTSTORE SETTINGS' section with sub-sections: Settings (General, Localization, Configuration Settings), Integrations (Shipping, Payments), SSO, Promotions, Service Management, and Appearance (Themes, Layouts). A callout '2' points to the 'Themes' link in the Appearance section. The main content area is titled 'Themes for DefaultStore' and contains a table with columns 'THEME' and 'ACTION'. The table lists three themes: 'ccrzqard\_CC\_Theme\_Capricorn', 'ccrzqard\_CC\_Theme\_Capricorn\_Classic', and 'ccrzqard\_CC\_Theme\_Minimum', each with 'View', 'Download', and 'Enable' buttons. Below this is the 'Dynamic Category Extension Settings for DefaultStore' section with a 'Dynamic Category Extension API Class' input field containing 'cc\_hk\_Category'. Below that is the 'Dynamic Menu Extension Settings for DefaultStore' section with a 'Dynamic Menu Extension API Class' input field containing 'cc\_hk\_Menu'. At the bottom is the 'User Interface Extension Settings for DefaultStore' section with a 'User Interface Extension API Class' input field containing 'c.cc\_example\_hk\_UserInterface'. A callout '3' points to this input field. Each extension settings section has a 'Save' button and a note: 'Note: To reference a class defined in a subscriber org, prefix the class name with 'c.''

 **Tip:** For more help with creating an extension, including considerations and conventions, see [Extend a Global Extension Point Class](#).

## Methods

### IN THIS SECTION:

#### [headContent](#)

Calls other inner methods of the `ccrz.cc_hk_UserInterface` class for populating the default JavaScript and CSS libraries referenced in each storefront page's `<head>` element.

#### [metaContent](#)

Returns a default definition for the `<meta name="viewport">` element included in each storefront page's `<head>` element.

#### [resourcePath](#)

Returns the relative URL path from the storefront to a JavaScript or CSS file included in a static resource that the managed package installs. Other methods of the `ccrz.cc_hk_UserInterface` class reference this path when constructing `<script>` or `<link>` elements for inclusion in each page's `<head>`.



### [themeResourcePath](#)

Returns the relative URL path from the storefront to a JavaScript or CSS file included a static resource for the storefront's enabled theme. Other methods of the `ccrz.cc_hk_UserInterface` class reference this path when constructing `<script>` or `<link>` elements for inclusion in each page's `<head>`.

### [standardIncludes](#)

Returns `<script>` and `<link>` elements that reference the default third-party JavaScript libraries and CSS files, respectively. These libraries are referenced in each storefront page's `<head>` element.

### [ccrzIncludes](#)

Returns `<script>` elements that reference proprietary JavaScript libraries or heavily modified third-party libraries that are essential to default storefront functionality. These libraries are referenced in each storefront page's `<head>` element.

### [pageConfigCSS3Includes](#)

Returns `<link>` elements that reference custom CSS files used with a Bootstrap 3-enabled storefront theme. These files are referenced in each storefront page's `<head>` element.

### [pageConfigCSSIncludes](#)

Returns `<link>` elements that reference custom CSS files used with a Bootstrap 2.3.2-enabled storefront theme. These files are referenced in each storefront page's `<head>` element.

### [standardRespondJS](#)

Returns a `<script>` element that specifies the path to the third-party `respond.js` library, which is used with a Bootstrap 2.3.2-enabled storefront theme. This library is referenced in each storefront page's `<head>` element.

### [pageConfigJS3Includes](#)

Returns `<script>` elements that reference custom JavaScript libraries used with a Bootstrap 3-enabled storefront theme. These libraries are referenced in each storefront page's `<head>` element.

### [pageConfigJSIncludes](#)

Returns `<script>` elements that reference custom JavaScript libraries used with a Bootstrap 2.3.2-enabled storefront theme. These libraries are referenced in each storefront page's `<head>` element.

### [standardUIProperties](#)

Returns `<script>` elements that reference the default `uiproperties.js` file and any theme-specific override of this file, if applicable. The `uiproperties.js` file defines default Handlebars templates, CSS selectors, and other properties for rendering storefront components. These files are referenced in each storefront page's `<head>` element.

### [endContent](#)

Defines `<script>` elements that reference third-party JavaScript libraries for a Bootstrap 3-enabled storefront theme. These libraries are referenced just before the HTML Body Include End page include.

### [canonicalUrlInclude](#)

Returns the canonical and alternate language links for the Home page, Product List page, and Product Detail page.

### [getInstance](#)

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_UserInterface` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.


## `ccrz.cc_hk_UserInterface.headContent`

Calls other inner methods of the `ccrz.cc_hk_UserInterface` class for populating the default JavaScript and CSS libraries referenced in each storefront page's `<head>` element.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method returns a value for either a Bootstrap 3- or Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is either `boot3` or `classic`. We recommend that new solutions always use `boot3`.

## Signature

```
global virtual Map<String, Object> headContent (Map<String, Object>)
```

## Inputs

None

## Outputs

`Map<String, Object>` that includes the following key. The exact content depends on which Bootstrap version your storefront uses:

### **ccrz.cc\_hk\_UserInterface.HEAD\_CONTENT**

String that contains the HTML content to add to the page's `<head>` element. This content is added to the page source after the Head Include Begin and before the JavaScript that specifies page labels:

```
<head>
  <!--Head Include Begin-->
  <apex:include id="hib" pageName="{!headerBeginPage}" rendered="{!useHeaderBegin}"
/>

  <!--Core objects definitions-->
  <script type="text/javascript">
    var CCRZ = CCRZ || {};
    CCRZ.devMode = CCRZ.devMode||{!storefront.DevMode__c};
  </script>

  <!--ccrz.cc_hk_UserInterface.HEAD_CONTENT inserted here-->

  <!--Internal component that creates the JavaScript page labels-->
  <c:cc_PageLabelsInclude />

  <!--Head Include End-->
```

```
<apex:include id="hie" pageName="{!headerEndPage}" rendered="{!useHeaderEnd}" />
</head>
```

For a Bootstrap 3-enabled storefront, this method calls the following inner methods for populating the specific `<link>` and `<script>` elements to add:

1. `metaContent`
2. `standardIncludes`
3. `pageConfigCSS3Includes`
4. `standardRespondJS`
5. `pageConfigJS3Includes`
6. `standardUIProperties`
7. `canonicalUrlInclude`

For a Bootstrap 2.3.2-enabled storefront, this method calls the following inner methods for populating the specific `<link>` and `<script>` elements to add:

1. `metaContent`
2. `standardIncludes`
3. `pageConfigCSSIncludes`
4. `standardRespondJS`
5. `pageConfigJSIncludes`
6. `standardUIProperties`
7. `canonicalUrlInclude`



**Tip:** With this design, you can limit your extension to just the method that controls the portion of code you want to replace. For example, to customize only the `<meta>` content, you can override just the `ccrz.cc_hk_UserInterface.metaContent` method instead of overriding the entire `ccrz.cc_hk_UserInterface` class.

SEE ALSO:

[Salesforce Help Default JavaScript Libraries for Storefront Pages](#)

## ccrz.cc\_hk\_UserInterface.metaContent


Returns a default definition for the `<meta name="viewport">` element included in each storefront page's `<head>` element.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method returns a value for either a Bootstrap 3- or Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is either `boot3` or `classic`. We recommend that new solutions always use `boot3`.

## Signature

```
global virtual String metaContent()
```

## Inputs

None

## Outputs

String that specifies the `<meta name="viewport">` content, depending on which Bootstrap version your storefront uses:


### **ccrz.cc\_hk\_UserInterface.BS3\_STANDARD\_META\_VIEWPORT**

String that specifies default `<meta>` element values when the storefront uses the Bootstrap 3 framework:

```
<meta charset="utf-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1">
```

### **ccrz.cc\_hk\_UserInterface.STANDARD\_META\_VIEWPORT**

String that specifies default `<meta>` element values when the storefront uses the Bootstrap 2.3.2 framework:

 **Important:** The Bootstrap 2.3.2 library is intended only for compatibility with solutions that are upgrading from older versions of the managed package. We recommend that new solutions always use the Bootstrap 3 library.

```
<meta name="viewport" content="width=device-width, initial-scale=1.0, user-scalable=no" />
```

SEE ALSO:

[Salesforce Help Default JavaScript Libraries for Storefront Pages](#)

[Mozilla Developers <meta>: The Document-level Metadata element](#)


## ccrz.cc\_hk\_UserInterface.resourcePath

Returns the relative URL path from the storefront to a JavaScript or CSS file included in a static resource that the managed package installs. Other methods of the `ccrz.cc_hk_UserInterface` class reference this path when constructing `<script>` or `<link>` elements for inclusion in each page's `<head>`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method returns a value for either a Bootstrap 3- or Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is either `boot3` or `classic`. We recommend that new solutions always use `boot3`.

## Signature

```
global static String resourcePath(String, String)
```


## Inputs

### resourceName

String that specifies the name of a static resource that contains JavaScript or CSS files, such as the `CC_JavaScript_Framework` or `CCRZ_JS` static resources that the managed package installs.

### elementPath

String that specifies the path to a specific file within the `resourceName`.

 **Note:** This string can't begin with a forward slash character (/).

## Outputs

String that specifies the relative path from the storefront to a file in the specified static resource, prefixed with the appropriate namespace, such as `ccrz`.

### Example: Example

The `CCRZ_JS` static resource that the managed package installs includes several files, such as `bootstrap.css` and `jquery.js`. To construct paths to these files, other methods of the `ccrz.cc_hk_UserInterface` class invoke the `resourcePath` method:

```
resourcePath(CCRZ_JS, 'v004/boot3/css/bootstrap.css')
resourcePath(CCRZ_JS, 'v004/boot3/jquery-3.5.1.js')
```

In a `<link>` or `<script>` element, these values output as:

```
/DefaultStore/resource/1590503075000/ccrz__CCRZ_JS/v004/boot3/css/bootstrap.css
/DefaultStore/resource/1590503075000/ccrz__CCRZ_JS/v004/boot3/jquery-3.5.1.js
```

## SEE ALSO:

[Salesforce Help Style Your Storefront with a Theme](#)


## ccrz.cc\_hk\_UserInterface.themeResourcePath

Returns the relative URL path from the storefront to a JavaScript or CSS file included a static resource for the storefront's enabled theme. Other methods of the `ccrz.cc_hk_UserInterface` class reference this path when constructing `<script>` or `<link>` elements for inclusion in each page's `<head>`.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method returns a value for either a Bootstrap 3- or Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is either `boot3` or `classic`. We recommend that new solutions always use `boot3`.


### Signature

```
global static String themeResourcePath(String)
```

### Inputs

#### elementPath

String that specifies the path to a specific file within the static resource for storefront's enabled theme.

 **Note:** This string can't begin with a forward slash character (/).

### Outputs

String that specifies the relative path from the storefront to a file in the theme's static resource, prefixed with the appropriate namespace, such as `ccrz`.

#### Example: Example

The `CC_Theme_Capricorn` static resource that the managed package installs includes several files, such as `styles.css` and an override of the default `uiproperties.js`. To construct paths to these files, other methods of the `ccrz.cc_hk_UserInterface` class invoke the `themeResourcePath` method:

```
themeResourcePath('css3/styles.css')
themeResourcePath('js/uiproperties.js')
```

In a `<link>` or `<script>` element, these paths output as:

```
/DefaultStore/resource/1590503075000/ccrz__CC_Theme_Capricorn/css3/styles.css
/DefaultStore/resource/1590503075000/ccrz__CC_Theme_Capricorn/js/uiproperties.js
```

SEE ALSO:

[Salesforce Help Style Your Storefront with a Theme](#)

## ccrz.cc\_hk\_UserInterface.standardIncludes

Returns `<script>` and `<link>` elements that reference the default third-party JavaScript libraries and CSS files, respectively. These libraries are referenced in each storefront page's `<head>` element.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



**Note:** This method returns a value for either a Bootstrap 3- or Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is either `boot3` or `classic`. We recommend that new solutions always use `boot3`.

### Signature

```
global virtual String standardIncludes()
```

### Inputs

None



**Note:** This method executes only when the [Override Standard Include storefront configuration setting](#) is `FALSE` (the default value).

### Outputs

String that specifies the `<script>` and `<link>` elements for default JavaScript libraries and CSS files, respectively. The exact string depends on:

- Which Bootstrap version your storefront uses.
- The value of the [Use Minified storefront configuration setting](#).


**Bootstrap 3 (default)**

When Use Minified is `TRUE` (its default value), this method returns the following string:


```
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/boot3/css/ccrz-libs.min.css" />
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/ccrz-libsh-min.js"></script>
```

When Use Minified is `FALSE`, this method returns the following string:

```
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/boot3/css/bootstrap-datepicker3.css" />
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/boot3/css/font-awesome.min.css" />
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/boot3/css/jquery-ui-custom.css" />
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/boot3/css/bootstrap.css" />
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/boot3/css/shimbs3.css" />
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/boot3/css/nouislider.css" />
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/jquery-3.5.1.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/jquery-migrate-3.3.0.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/jquery-ui-custom.1.13.2.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/jquery.validate.1.19.5.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/underscore-1.13.1.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/backbone-1.4.0.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/accounting-0.4.2.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/handlebars-v4.7.7.js"></script>
```

 **Note:** References to other default libraries for a Bootstrap 3-enabled storefront are returned from the `ccrz.cc_hk_UserInterface.endContent` method.

**Bootstrap 2.3.2**

 **Important:** The Bootstrap 2.3.2 library is intended only for compatibility with solutions that are upgrading from older versions of the managed package. We recommend that new solutions always use the Bootstrap 3 library.


When Use Minified is `TRUE`, this method returns the following string:


```
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/ccrz-lib-min.js"></script>
```



When Use Minified is FALSE, this method returns the following string:

```
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/jquery-3.5.1.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/jquery-migrate-3.3.0.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/jquery-ui-custom.1.13.2.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/jquery.cookie-1.4.1.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/jquery.hoverIntent-1.10.1.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/jquery-vertical-accordion-menu/js/jquery.dcjqaccordion.2.7.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/jquery.validate.1.19.5.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/underscore-1.13.1.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/backbone-1.4.0.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/modernizr-2.8.3.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/accounting-0.4.2.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/handlebars-v4.7.7.js"></script><
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/js-image-slider-2016.9.27.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/bootstrap-datepicker-1.9.0.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/js2form.2011.9.19.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/form2js-2010.9.9.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/backbone.paginator.2.0.8.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/bootstrap-2.3.2/js/bootstrap.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/nouislider.14.6.2.js"></script>
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/classic/css/bootstrap-datepicker.css" />
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/classic/css/jquery-ui-custom.css" />
<link type="text/css" rel="stylesheet"
href="CCRZ_JS_resourcePath/v004/classic/css/nouislider.css" />
```

 **Note:** The `CCRZ_JS_resourcePath` variable represents the relative path from the storefront to the namespaced `CCRZ_JS` static resource that the managed package installs, such as `/DefaultStore/resource/1590503075000/ccrz__CCRZ_JS`. The `ccrz.cc_hk_UserInterface.resourcePath` method returns each file's path in the static resource.

 **Note:** The specific libraries and versions of those libraries that are referenced vary depending on your version of the managed package, including patch releases. For more information, see [Default JavaScript Libraries for Storefront Pages](#).

SEE ALSO:

[Salesforce Help Default JavaScript Libraries for Storefront Pages](#)

[Salesforce Help Exclude the Standard CSS and JavaScript Libraries](#)


## ccrz.cc\_hk\_UserInterface.ccrzIncludes

Returns `<script>` elements that reference proprietary JavaScript libraries or heavily modified third-party libraries that are essential to default storefront functionality. These libraries are referenced in each storefront page's `<head>` element.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8


 **Note:** This method returns a value for either a Bootstrap 3- or Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is either `boot3` or `classic`. We recommend that new solutions always use `boot3`.

### Signature

```
global virtual String ccrzIncludes ()
```

### Inputs

None

 **Note:** This method is called within the `ccrz.cc_hk_UserInterface.standardIncludes` method for appending these libraries to the other standard JavaScript and CSS libraries.

### Outputs

String that specifies `<script>` elements for the JavaScript libraries to include. The exact string depends on:

- Which Bootstrap version your storefront uses.
- The value of the [Use Minified storefront configuration setting](#).

**Bootstrap 3 (default)**


When Use Minified is `TRUE` (its default value), this method returns the following string:

```
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/cc3-min.js"></script>
```

When Use Minified is `FALSE`, this method returns the following string:

```
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/jquery.currency.js"></script>
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/cloudcraze.js"></script>
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/CC_Backbone.js"></script>
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/backbone_models.js"></script>
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/analytics.js"></script>
```

**Bootstrap 2.3.2**


 **Important:** The Bootstrap 2.3.2 library is intended only for compatibility with solutions that are upgrading from older versions of the managed package. We recommend that new solutions always use the Bootstrap 3 library.

When Use Minified is `TRUE`, this method returns the following string:

```
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/cc-min.js"></script>
```

When Use Minified is `FALSE`, this method returns the following string:

```
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/jquery.currency.js"></script>
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/controllers_new_slider.js"></script>
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/cloudcraze.js"></script>
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/CC_Backbone.js"></script>
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/backbone_models.js"></script>
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/analytics.js"></script>
```

 **Note:** The `CC_JavaScript_Framework_resourcePath` variable represents the relative path from the storefront to the namespaced `CC_JavaScript_Framework` static resource that the managed package installs, such as `/DefaultStore/resource/1590503075000/ccrz__CC_Javascript_Framework`. The `ccrz.cc_hk_UserInterface.resourcePath` method returns each file's path in the static resource.

SEE ALSO:

[CC\\_Backbone.js](#)

[cloudcraze.js](#)


## ccrz.cc\_hk\_UserInterface.pageConfigCSS3Includes

Returns `<link>` elements that reference custom CSS files used with a Bootstrap 3-enabled storefront theme. These files are referenced in each storefront page's `<head>` element.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method returns a value only for a Bootstrap 3-enabled storefront, where the [Template Version storefront configuration setting](#) is `boot3` (its default value). We recommend that new solutions always use `boot3`.

### Signature

```
global static String pageConfigCSS3Includes ()
```


### Inputs

None

### Outputs

String that specifies `<link>` elements for each CSS file to include:

```
<link type="text/css" rel="stylesheet" href="themeResourcePath/relativePathToCSSFile" />
```

 **Note:** The `themeResourcePath` variable represents the relative path from the storefront to the namespaced static resource for the storefront's enabled theme, such as `/DefaultStore/resource/1590503075000/ccrz__CC_Theme_Capricorn`. [ccrz.cc\\_hk\\_UserInterface.themeResourcePath method](#) returns each file's path in the static resource.

Each CSS file also requires a corresponding CSS 3 Include configuration in your storefront's configuration settings that specifies this relative path. For help with setting up a theme static resource and defining its CSS paths, see [Style Your Storefront with a Theme](#).

### Example: Example

The managed package installs a `CC_Theme_Capricorn` static resource that contains files for the Capricorn Coffee demo storefront. That static resource contains a `css3` directory, which contains a `styles.css` file. The demo storefront specifies a CSS 3 Include

configuration setting, with a value of `css3/styles.css`. This method then returns the following `<link>` definition for that CSS file:

```
<link type="text/css" rel="stylesheet"
href="/DefaultStore/resource/1590503075000/ccrz__CC_Theme_Capricorn/css3/styles.css"
/>
```

SEE ALSO:

[Salesforce Help Style Your Storefront with a Theme](#)


## ccrz.cc\_hk\_UserInterface.pageConfigCSSIncludes

Returns `<link>` elements that reference custom CSS files used with a Bootstrap 2.3.2-enabled storefront theme. These files are referenced in each storefront page's `<head>` element.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Important:** This method returns a value only for a Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is `classic`. This configuration value is intended only for compatibility with solutions that are upgrading from older versions of the managed package. We recommend that new solutions always use `boot3` instead.

### Signature

```
global static String pageConfigCSSIncludes()
```


### Inputs

None

### Outputs

String that specifies `<link>` elements for each CSS file to include:

```
<link type="text/css" rel="stylesheet" href="themeResourcePath/relativePathToCSSFile />
```

 **Note:** The `themeResourcePath` variable represents the relative path from the storefront to the namespaced static resource for the storefront's enabled theme, such as `/DefaultStore/resource/1590503075000/ccrz__CC_Theme_Capricorn`. `ccrz.cc_hk_UserInterface.themeResourcePath` method returns each file's path in the static resource.

For help with setting up a theme static resource and defining its CSS paths, see [Style Your Storefront with a Theme](#).


## ccrz.cc\_hk\_UserInterface.standardRespondJS

Returns a `<script>` element that specifies the path to the third-party `respond.js` library, which is used with a Bootstrap 2.3.2-enabled storefront theme. This library is referenced in each storefront page's `<head>` element.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Important:** This method returns a value only for a Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is `classic`. This configuration value is intended only for compatibility with solutions that are upgrading from older versions of the managed package. We recommend that new solutions always use `boot3` instead.

### Signature

```
global virtual String standardRespondJS (String)
```


### Inputs

None

### Outputs

String that specifies a `<script>` element for the JavaScript library to include:

```
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/classic/respond-1.4.2.js"></script>
```

 **Note:** The `themeResourcePath` variable represents the relative path from the storefront to the namespaced static resource for the storefront's enabled theme, such as `/DefaultStore/resource/1590503075000/ccrz__CC_Theme_Capricorn`. `ccrz.cc_hk_UserInterface.themeResourcePath` method returns each file's path in the static resource.

For a Bootstrap 3-enabled storefront, this method returns an empty string.


## ccrz.cc\_hk\_UserInterface.pageConfigJS3Includes

Returns `<script>` elements that reference custom JavaScript libraries used with a Bootstrap 3-enabled storefront theme. These libraries are referenced in each storefront page's `<head>` element.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method returns a value only for a Bootstrap 3-enabled storefront, where the [Template Version storefront configuration setting](#) is `boot3` (its default value). We recommend that new solutions always use `boot3`.

### Signature

```
global static String pageConfigJS3Includes()
```


### Inputs

None

### Outputs

String that specifies `<script>` elements for each JavaScript file to include:

```
<script type="text/javascript"
src="themeResourcePath/relativePathToJavaScriptFile"></script>
```

 **Note:** The `themeResourcePath` variable represents the relative path from the storefront to the namespaced static resource for the storefront's enabled theme, such as `/DefaultStore/resource/1590503075000/ccrz__CC_Theme_Capricorn`. [ccrz.cc\\_hk\\_UserInterface.themeResourcePath](#) method returns each file's path in the static resource.

## ccrz.cc\_hk\_UserInterface.pageConfigJSIncludes

Returns `<script>` elements that reference custom JavaScript libraries used with a Bootstrap 2.3.2-enabled storefront theme. These libraries are referenced in each storefront page's `<head>` element.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Important:** This method returns a value only for a Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is `classic`. This configuration value is intended only for compatibility with solutions that are upgrading from older versions of the managed package. We recommend that new solutions always use `boot3` instead.

## Signature

```
global static String pageConfigJSIncludes()
```

## Inputs

None

## Outputs

String that specifies `<script>` elements for each JavaScript file to include:

```
<script type="text/javascript"
src="themeResourcePath/relativePathToJavaScriptFile></script>
```

**Note:** The `themeResourcePath` variable represents the relative path from the storefront to the namespaced static resource for the storefront's enabled theme, such as `/DefaultStore/resource/1590503075000/ccrz__CC_Theme_Capricorn`. [ccrz.cc\\_hk\\_UserInterface.themeResourcePath](#) method returns each file's path in the static resource.

**Important:** Each JavaScript file also requires a corresponding page configuration that specifies this relative path within the static resource. Page configurations are deprecated in all versions of the managed package that are currently supported.

## ccrz.cc\_hk\_UserInterface.standardUIProperties


Returns `<script>` elements that reference the default `uiproperties.js` file and any theme-specific override of this file, if applicable. The `uiproperties.js` file defines default Handlebars templates, CSS selectors, and other properties for rendering storefront components. These files are referenced in each storefront page's `<head>` element.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method returns a value for either a Bootstrap 3- or Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is either `boot3` or `classic`. We recommend that new solutions always use `boot3`.

## Signature

```
global virtual String standardUIProperties()
```

## Inputs

None

## Outputs


String that specifies `<script>` elements that reference `uiproperties.js`. The exact string depends on the value of the [Use Minified storefront configuration setting](#):

- When Use Minified is `TRUE` (its default value), this method returns the following string:

```
<script type="text/javascript" src="themeResourcePath/js/uiproperties.js />
```

- When Use Minified is `FALSE`, this method returns the following string:

```
<script type="text/javascript"
src="CC_JavaScript_Framework_resourcePath/js/uiproperties.js />
<script type="text/javascript" src="themeResourcePath/js/uiproperties.js />
```

 **Note:** The `CC_JavaScript_Framework_resourcePath` variable represents the relative path from the storefront to the namespaced `CC_JavaScript_Framework` static resource that the managed package installs, such as `/DefaultStore/resource/1590503075000/ccrz__CC_Javascript_Framework`. The [ccrz.cc\\_hk\\_UserInterface.resourcePath](#) method returns each file's path in the static resource.

 **Note:** The `themeResourcePath` variable represents the relative path from the storefront to the namespaced static resource for the storefront's enabled theme, such as

`/DefaultStore/resource/1590503075000/ccrz__CC_Theme_Capricorn.  
ccrz.cc_hk_UserInterface.themeResourcePath` method returns each file's path in the static resource.

SEE ALSO:

[uiproperties.js](#)

[Salesforce Help Style Your Storefront with a Theme](#)


## ccrz.cc\_hk\_UserInterface.endContent

Defines `<script>` elements that reference third-party JavaScript libraries for a Bootstrap 3-enabled storefront theme. These libraries are referenced just before the HTML Body Include End page include.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Note:** This method returns a value only for a Bootstrap 3-enabled storefront, where the [Template Version storefront configuration setting](#) is `boot3` (its default value). We recommend that new solutions always use `boot3`.

### Signature

```
global virtual Map<String, Object> endContent (Map<String, Object>)
```

### Inputs

None

### Outputs

`Map<String, Object>` that can include the following keys:

#### **ccrz.cc\_hk\_UserInterface.EOB\_CONTENT**



String that contains the `<script>` definitions to insert before the HTML Body Include End page include. The exact string depends on the value of the [Use Minified storefront configuration setting](#):

- When Use Minified is `TRUE` (its default value), this method returns the following string:

```
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/ccrz-libse-min.js"></script>
```

- When Use Minified is `FALSE`, this method returns the following string:

```
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/bootstrap-datepicker-1.9.0.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/js2form.2011.9.19.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/form2js-2010.9.9.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/backbone.paginator.2.0.8.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/nouislider.14.1.1.js"></script>
<script type="text/javascript"
src="CCRZ_JS_resourcePath/v004/boot3/bootstrap.3.4.1.js"></script>
```

-  **Note:** The `CCRZ_JS_resourcePath` variable represents the relative path from the storefront to the namespaced `CCRZ_JS` static resource that the managed package installs, such as `/DefaultStore/resource/1590503075000/ccrz__CCRZ_JS`. The [ccrz.cc\\_hk\\_UserInterface.resourcePath](#) method returns each file's path in the static resource.
-  **Note:** The specific libraries and versions of those libraries that are referenced vary depending on your version of the managed package, including patch releases. For more information, see [Default JavaScript Libraries for Storefront Pages](#).

SEE ALSO:

[Salesforce Help Default JavaScript Libraries for Storefront Pages](#)


## ccrz.cc\_hk\_UserInterface.canonicalUrlInclude

Returns the canonical and alternate language links for the Home page, Product List page, and Product Detail page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10


-  **Note:** This method returns a value for either a Bootstrap 3- or Bootstrap 2.3.2-enabled storefront, where the [Template Version storefront configuration setting](#) is either `boot3` or `classic`. We recommend that new solutions always use `boot3`.

## Signature

```
global virtual String canonicalUrlInclude()
```

## Inputs

None

 **Note:** This method executes only when the [Enable Canonical and Alternate Language Links storefront configuration setting](#) is `TRUE`.


## Outputs

String that specifies the current page's canonical and alternate language links:

### Canonical link

The authoritative path of the Home page, Product List page, or Product Detail page to present to search engine bots. This URL includes the default storefront locale for a guest user, such as `en_US`.

```
<link rel="canonical"
href="https://www.mydomain.com/DefaultStore/ccrz__ProductList?categoryId=an18digitCategoryId&cclcl=en_US"
/>
```

 **Note:** If your storefront uses friendly URLs, the canonical URL for the Product List page and Product Detail page uses the friendly URL values.

### Alternate language link

The authoritative path of Home page, Product List page, or Product Detail page for each locale that your storefront supports, including the default locale. These locale-specific URLs help search engine bots index the page for users in each locale.

```
<link rel="alternate" hreflang="en-us"
href="https://www.mydomain.com/DefaultStore/ccrz__ProductList?categoryId=an18digitCategoryId&cclcl=en_US"
/>
```

```
<link rel="alternate" hreflang="ja-jp"
href="https://www.mydomain.com/DefaultStore/ccrz__ProductList?categoryId=an18digitCategoryId&cclcl=ja_JP"
/>
```

SEE ALSO:

- [Salesforce Help Canonical URLs for Storefront Pages](#)
- [Salesforce Help Alternate Language Links for Storefront Pages](#)
- [Salesforce Help Insert Canonical URLs and Alternate Language URLs](#)
- [Salesforce Help Serve Friendly URLs for Storefront Pages](#)

## ccrz.cc\_hk\_UserInterface.getInstance

Returns an instance of this class configured for the storefront, such as the default `ccrz.cc_hk_UserInterface` class. If you create subscriber code that extends the default class, this method returns an instance of your custom class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static ccrz.cc_hk_UserInterface getInstance(Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.cc_hk_UserInterface.PARAM_STOREFRONT`

String that specifies the name of a storefront.



**Note:** If the input data doesn't include this key, the method falls back to `ccrz.cc_CallContext.storefront`.

## Outputs

This method returns an instance of `ccrz.cc_hk_UserInterface`, or your custom extension point class.

# Utility Classes for B2B Commerce for Visualforce

B2B Commerce for Visualforce provides several Apex classes with static fields, which effectively behave like global objects that expose certain storefront session data. Other classes and methods in the managed package can easily reference these static fields.



**Note:** Sometimes, B2B Commerce for Visualforce updates utility classes in a new version of the B2B Commerce managed package. In subscriber code, you reference the latest version of these classes that's available in your version of the managed package.

### IN THIS SECTION:

#### `ccrz.cc_bean_CartSummary`

Represents a `ccrz__E_Cart__c` record.

#### `ccrz.cc_bean_Category`

Represents a `ccrz__E_Category__c` record.

#### `ccrz.cc_bean_EffectiveAccount`

Represents an effective account that a buyer can access other than the buyer's default assigned account. The methods of the `ccrz.cc_hk_EffectiveAccount` extension point class create and reference instances of this class.

[ccrz.cc\\_bean\\_MenuItem](#)

Represents either a `ccrz__E_Menu__c` or `ccrz__E_MenuItem__c` record, depending on which level of menu the object corresponds to.

[ccrz.cc\\_bean\\_Message](#)

Represents an error message or other text string to pass from the server and show on a storefront page.

[ccrz.cc\\_bean\\_MockContactAddress](#)

Represents a `ccrz__E_ContactAddr__c` record.

[ccrz.cc\\_bean\\_MockOrder](#)

Represents a `ccrz__E_Order__c` record.

[ccrz.cc\\_bean\\_PromoRD](#)

Passes promotion details between other global API method calls and returns JSON data for showing promotions on the storefront.

[ccrz.cc\\_ctrl\\_hlpr\\_ShippingOption](#)

Represents a shipping option available in the Shipping Method picklist on the storefront's Checkout page.

[ccrz.cc\\_CallContext](#)

Provides a standard interface for referencing the current user, storefront activity, and other session information required for B2B Commerce for Visualforce API calls.

[ccrz.cc\\_RemoteActionContext](#)

Initializes `ccrz.cc_CallContext` from an Apex remote action.

[ccrz.cc\\_RemoteActionResult](#)

Returns the result of an Apex remote action.

[ccrz.ccMultiListIterator](#)

Iterates through multiple Apex lists and creates `List<Object>` representations. This class is used with batch jobs where the batched data can't be constructed with SOQL.

[ccrz.ccMultiSetIterator](#)

Iterates through multiple Apex sets and creates `List<Object>` representations. This class is used with batch jobs where the batched data can't be constructed with SOQL.

[ccrz.ccPaymentProcessor](#)

Provides an extension point class for processing a specific payment type for all payment flows, including checkout, subscriptions, and invoices.

[ccrz.ccProductIndexBuilder](#)

Builds product index records by executing a series of synchronously executed Apex batch jobs. Each batch job includes a `finish` method that calls back into the `ccrz.ccProductIndexBuilder` class for starting the next batch job.

[ccrz.ccSiteIndexBuilder](#)

Builds site index records by executing a series of synchronously executed Apex batch jobs. Each batch job includes a `finish` method that calls back into the `ccrz.ccSiteIndexBuilder` class for starting the next batch job.

[ccrz.ccUtil](#)

Defines a set of static utility methods for evaluating, parsing, or hashing B2B Commerce for Visualforce object data.

## ccrz.cc\_bean\_CartSummary

Represents a `ccrz__E_Cart__c` record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Properties

This class defines the following Apex [global](#) properties. For more information about Apex properties and accessors, see [Apex Properties](#).

### **adjustment {get; set;}**

Decimal that specifies the amount to adjust the order price by after the order is placed.

### **allowCheckout {get; set;}**

Boolean

Value	Usage
<i>true</i>	The buyer is allowed to check out with this cart.
<i>false</i>	The buyer isn't allowed to check out with this cart.

### **billingAddress {get;}**

[ccrz.cc\\_bean\\_MockContactAddress](#) that represents the billing address associated with the cart.

### **buyerContactId {get;}**

ID of the contact associated with the cart's assigned user.

### **buyerEmail {get;}**

String that specifies the buyer's email address.

### **buyerFirstName {get;}**

String that specifies the buyer's first name.

### **buyerLastName {get;}**

String that specifies the buyer's last name.

### **buyerPhone {get;}**

String that specifies the buyer's phone number.

### **cartItems {get; set;}**

`List<ccrz.cc_bean_CartItem>` of the items in the cart.

### **ccEmailAddresses {get;}**

String that specifies email addresses to include in the CC address line of the order confirmation email.

**contractId {get;}**

ID of the contract ID when the cart's payment method is Contract.

**couponName {get; set;}**

String that specifies the name of a coupon applied to the cart.

**encryptedId {get;}**

String that specifies a semi-random UUID to refer to the cart in page URLs.

**extrinsic {get; set;}**

Map<String, Object> that represents a placeholder for external data that you can pass to this bean and other objects that refer to this bean.

**grossTotal {get; set;}**

Decimal that specifies the total cost calculated for the cart.

**hasCoupon {get;}**

Boolean

Value	Usage
<i>true</i>	A coupon is applied to the cart.
<i>false</i>	No coupon has been applied to the cart.

**isActive {get;}**

Boolean

Value	Usage
<i>true</i>	The cart is <i>active</i> for the buyer's storefront session. Only one cart can be active per account at one time.
<i>false</i>	The cart is <i>inactive</i> for the buyer's storefront session.



**Note:** For more information about active and inactive carts, see [Considerations for Supporting a Buyer's Cart Experience](#).

**isValidated {get;}**

Boolean

Value	Usage
<i>true</i>	The cart has been validated.
<i>false</i>	The cart hasn't been validated.

**lastModifiedDateStr {get;}**

String representation of the date when the cart was most recently modified.

**messages**

List<ccrz.cc\_bean\_Message> of messages related to the cart.



**mockContact {get;}**

`ccrz.cc_bean_MockContact` that represents the contact associated with the cart's assigned user.

**name {get;}**

String that specifies a name for the cart in the My Carts section of the My Account page.

**note {get; set;}**

String that specifies a shipping note entered during checkout.

**payments {get;}**

`List<ccrz.cc_bean_Payment>` of the cart's available payment methods.

**paymentType {get; set;}**

String that specifies the cart's selected payment method.

**poNumber {get;}**

String that specifies the purchase order (PO) number entered during checkout.

**preventBillingEdit {get;}**

Boolean

Value	Usage
<i>true</i>	The buyer isn't allowed to edit their billing address on the storefront.
<i>false</i>	The buyer is allowed to edit their billing address on the storefront.

**preventContactEdit {get;}**

Boolean

Value	Usage
<i>true</i>	The buyer isn't allowed to edit their contact information on the storefront.
<i>false</i>	The buyer is allowed to edit their contact information on the storefront.

**preventShippingEdit {get;}**

Boolean

Value	Usage
<i>true</i>	The buyer isn't allowed to edit their shipping address on the storefront.
<i>false</i>	The buyer is allowed to edit their shipping address on the storefront.

**requestedDate {get;}**

Date that specifies the delivery date that the buyer requested during checkout.

**requestedDateStr {get;}**

String representation of the `requestedDate`.

**sfid {get; set;}**

Salesforce 18-digit ID of the cart record.

**shipComplete {get;}**

Boolean

Value	Usage
<i>true</i>	The buyer selected the Ship Complete option during checkout.
<i>false</i>	The buyer didn't select the Ship Complete option during checkout.

**shippingAddress {get;}**`ccrz.cc_bean_MockContactAddress` that represents the shipping address associated with the cart.**shippingCharge {get;}**

Decimal that specifies the total shipping cost for the cart.

**shippingInst {get;}**

String that specifies any comments or notes associated with the selected shipping method.

**shippingMethod {get;}**

String that specifies the cart's selected shipping method.

**shippingMethods {get;}**List<`ccrz.cc_ctrl_hlpr_ShippingOption`> of the terms and conditions related to the cart.**subTotal {get; set;}**

Decimal that specifies the total entitled price of all items in the cart.

**surchargeAmount {get; set;}**

Decimal that specifies charges other than taxes or shipping applied to the cart.

**tax {get; set;}**

Decimal that specifies the tax amount calculated during checkout.

**terms {get; set;}**List<`ccrz.cc_bean_Terms`> of the terms and conditions related to the cart.**totalDiscount {get; set;}**

Decimal that specifies the total amount discounted from the cart.

**totalInfo {get;}**

String that specifies information about the cart. The managed package doesn't reference this field by default.

## Constructors

This class defines the following constructors:

**ccrz.cc\_bean\_CartSummary**

Default constructor that doesn't take any parameters and doesn't set any values. When you use this constructor, make sure that you set the individual fields on bean.

**ccrz.cc\_bean\_CartSummary(ccrz\_\_E\_Cart\_\_c cart)**

Constructor that sets fields on the `ccrz.cc_bean_Summary` from the passed-in cart record.

**ccrz.cc\_bean\_CartSummary(ccrz\_\_E\_Cart\_\_c cart, Boolean fullData)**

Constructor that sets fields on the `ccrz.cc_bean_Summary` from the passed-in cart record. The value of `fullData` determines how many fields are set:

Value	Usage
<i>true</i>	The constructor tries to set all fields on the bean.
<i>false</i>	The fields related to the buyer, shipping information, and billing information aren't set.

## Methods

This class doesn't define any methods.

SEE ALSO:

[ccrz\\_\\_E\\_Cart\\_\\_c](#)

[ccrz.cc\\_api\\_CartExtension](#)

[Salesforce Help: Considerations for Supporting a Buyer's Cart Experience](#)

[Salesforce Help: Offer Coupons for Discounts on Storefront Purchases](#)

## ccrz.cc\_bean\_Category

Represents a `ccrz__E_Category__c` record.

## Compatibility

This class is deprecated and used only with the legacy `ccrz.cc_hk_Category` extension point class for versions of CloudCraze earlier than 4.5.

## Properties

This class defines the following Apex `global` properties. For more information about Apex properties and accessors, see [Apex Properties](#).

### **children {get; set;}**

`List<ccrz.cc_bean_Category>` of subcategories for the current category. Each subcategory can specify its own subcategories.

### **formatter {get; set;}**

`ccrz__E_LocaleFormatter__c`

### **name {get; set;}**

String that specifies a category name.

### **sfid {get; set;}**

String that specifies the Salesforce 18-digit ID of the menu or menu item record.

## Constructors

This class defines the following constructors:

### **ccrz.cc\_bean\_Category**

Default constructor that doesn't take any parameters and doesn't set any values. When you use this constructor, make sure that you set the individual fields on bean.

**ccrz.cc\_bean\_Category(String sfid, String name)**

Constructor that sets fields on the `ccrz.cc_bean_MenuItem` from the passed-in values.

## Methods

This class doesn't define any methods.

SEE ALSO:

[ccrz\\_\\_E\\_Category\\_\\_c](#)

[ccrz.cc\\_hk\\_Category](#) (Deprecated)

[Salesforce Help Organize Products into Categories for B2B Commerce for Visualforce](#)

## ccrz.cc\_bean\_EffectiveAccount

Represents an effective account that a buyer can access other than the buyer's default assigned account. The methods of the `ccrz.cc_hk_EffectiveAccount` extension point class create and reference instances of this class.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Properties

This class defines the following Apex [global](#) properties. For more information about Apex properties and accessors, see [Apex Properties](#).

**accountNumber {get; set;}**

String that specifies the account number for the account record that corresponds to a selected effective account.

**accountType {get; set;}**

String that specifies the Type value for the account record that corresponds to a selected effective account.

**billingAddress {get; set;}**

[ccrz.cc\\_bean\\_MockContactAddress](#) that contains the selected effective account's default billing address details.

**deliveryDateData {get; set;}**

Map<String, Object> of delivery date data returned from the [ccrz.cc\\_api\\_DeliveryDate](#) extension point class, if applicable.

**extrinsic {get; set;}**

Map<String, Object> that represents a placeholder for external data that you can pass to this bean and other objects that refer to this bean.

**name {get; set;}**

String that specifies the name of the account record that corresponds to a selected effective account.

**nextDelivery {get; set;}**

String with the value of the Request Date field on the active shopping cart record associated with the selected effective account. This field is returned with `deliveryDateData` from the `ccrz.cc_api_DeliveryDate` extension point class, if applicable.

**sfid {get; set;}**

String that specifies the ID of this `ccrz.cc_bean_EffectiveAccount` instance.

**shippingAddress {get; set;}**

`ccrz.cc_bean_MockContactAddress` that contains the selected effective account's default shipping address details.

## Constructors

This class defines the following constructors:

**ccrz.cc\_bean\_EffectiveAccount**

Default constructor that doesn't take any parameters and doesn't set any values. When you use this constructor, make sure that you set the individual fields on bean.

**ccrz.cc\_bean\_EffectiveAccount(account)**

Constructor that sets data for the effective account from the corresponding account record, including shipping and billing address details directly from the account record.

**ccrz.cc\_bean\_EffectiveAccount(Account account, ccrz\_\_E\_ContactAddr\_\_c contactAddressShipping, ccrz\_\_E\_ContactAddr\_\_c contactAddressBilling)**

Constructor that sets data for the effective account from the corresponding account record and its related `ccrz__E_ContactAddr__c` records for shipping and billing address details.

SEE ALSO:

[ccrz.cc\\_bean\\_MockContactAddress](#)

[ccrz.cc\\_hk\\_EffectiveAccount](#)

[Salesforce Help Allow a Buyer to Purchase for Multiple Effective Accounts](#)

## ccrz.cc\_bean\_MenuItem

Represents either a `ccrz__E_Menu__c` or `ccrz__E_MenuItem__c` record, depending on which level of menu the object corresponds to.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Properties

This class defines the following Apex [global](#) properties. For more information about Apex properties and accessors, see [Apex Properties](#).

### **children {get; set;}**

List<cc\_bean\_MenuItem> of the menu items nested within a menu, or other child menu items nested within a menu item.

### **displayName {get; set;}**

String that specifies the localized text that appears on the storefront for the menu or menu item.

### **friendlyURL {get; set;}**

String that specifies the friendly URL value, if specified, for a menu or menu item that links to a category or product.

### **linkURL {get; set;}**

String that specifies the URL that the menu links to:

- If `mType` is Category, this value is the category ID.
- If `mType` is Product, this value is the product ID.
- If `mType` is URL, this value is the relative path to a Visualforce page on the storefront or an absolute path to an external website.
- If `mType` is None, this value is null.

### **menuId {get; set;}**

String that specifies the value for the `data-menuid` attribute of the HTML element that represents the menu or menu item.

### **mType {get; set;}**

String that specifies the destination for the menu's link. For link type values and use cases, see [Provide Intuitive Storefront Navigation with Menus](#).

### **openInNewWindow {get; set;}**

Boolean

Value	Usage
<i>true</i>	Clicking the menu opens the URL in a new browser window or tab. This field is evaluated only when <code>mType</code> is URL.
<i>false</i>	Clicking the menu opens the URL in a same browser window or tab.

### **sequence {get; set;}**

Integer that specifies the order of the menu within the navigation bar, or order of the menu item within the menu. Lower values display first.

### **sfid {get; set;}**

Salesforce 18-digit ID of the menu or menu item record.

## Constructors

This class defines the following constructors:

### **ccrz.cc\_bean\_MenuItem**

Default constructor that doesn't take any parameters and doesn't set any values. When you use this constructor, make sure that you set the individual fields on bean.

### **ccrz.cc\_bean\_MenuItem(String displayName, String linkURL, Boolean openInNewWindow)**

Constructor that sets fields on the `ccrz.cc_bean_MenuItem` from the passed-in values.

### **ccrz.cc\_bean\_MenuItem(String displayName, String linkURL, Boolean openInNewWindow, String mType, Decimal sequence)**

Constructor that sets fields on the `ccrz.cc_bean_MenuItem` from the passed-in values.

### **ccrz.cc\_bean\_MenuItem(String sfid, String displayName, String linkURL, Boolean openInNewWindow)**

Constructor that sets fields on the `ccrz.cc_bean_MenuItem` from the passed-in values.

### **ccrz.cc\_bean\_MenuItem(String sfid, String displayName, String linkURL, Boolean openInNewWindow, String mType, Decimal sequence)**

Constructor that sets fields on the `ccrz.cc_bean_MenuItem` from the passed-in values.

### **ccrz.cc\_bean\_MenuItem(String sfid, String displayName, String linkURL, Boolean openInNewWindow, String menuId)**

Constructor that sets fields on the `ccrz.cc_bean_MenuItem` from the passed-in values.

### **ccrz.cc\_bean\_MenuItem(String sfid, String displayName, String linkURL, Boolean openInNewWindow, String mType, Decimal sequence, String menuId)**

Constructor that sets fields on the `ccrz.cc_bean_MenuItem` from the passed-in values.

## Methods

This class defines the following methods:

### **compareTo(ccrz.cc\_bean\_MenuItem compareTo)**

Compares the current `ccrz.cc_bean_MenuItem` object's `sequence` value to another `ccrz.cc_bean_MenuItem` object's `sequence` value. This method can return the following integer values:

Return Value	Usage
-1	The comparison had either of the following results: <ul style="list-style-type: none"> <li>The current <code>ccrz.cc_bean_MenuItem</code> object's <code>sequence</code> value is less than the compared object's <code>sequence</code> value.</li> <li>The current <code>ccrz.cc_bean_MenuItem</code> object's <code>sequence</code> value is null, but the compared object's <code>sequence</code> value isn't null.</li> </ul>
0	The comparison had either of the following results: <ul style="list-style-type: none"> <li>The current <code>ccrz.cc_bean_MenuItem</code> object's <code>sequence</code> value equals the compared object's <code>sequence</code> value.</li> <li>The <code>sequence</code> value of both objects is null.</li> </ul>

Return Value	Usage
1	<p>The comparison had either of the following results:</p> <ul style="list-style-type: none"> <li>The current <code>ccrz.cc_bean_MenuItem</code> object's <code>sequence</code> value is greater than the compared object's <code>sequence</code> value.</li> <li>The current <code>ccrz.cc_bean_MenuItem</code> object's <code>sequence</code> value isn't null, but the compared object's <code>sequence</code> value is null.</li> </ul>

SEE ALSO:

[ccrz\\_\\_E\\_Menu\\_\\_c](#)

[ccrz\\_\\_E\\_MenuItem\\_\\_c](#)

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

[Salesforce Help Serve Friendly URLs for Storefront Pages](#)

## ccrz.cc\_bean\_Message

Represents an error message or other text string to pass from the server and show on a storefront page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Properties

This class defines the following Apex `global` properties. For more information about Apex properties and accessors, see [Apex Properties](#).

#### **classToAppend {get; set;}**

String that specifies an HTML class name to add to the element where the message displays, such as an error section in a specific component. Make sure to specify this value when you create subscriber code that return `ccrz.cc_bean_Message` objects.

#### **displayClass {get; set;}**

String that specifies an extra HTML class name to add to the element where the message displays. This class name overrides the default class name returned for the message's `severity` value, if specified.

#### **labelId {get; set;}**

String that specifies the ID of an optional page label record that contains the message text to show, instead of referencing the `ccrz.cc_bean_Message` object's `message` field.



**message {get; set;}**

String that specifies the message to show.

**messageId {get; set;}**

String that specifies an optional ID for the message instance. This value isn't required to be unique.

**refersTo {get; set;}**

ID of a record that the message applies to. The managed package doesn't reference this field in any default messages, but is available if you want to reference a specific record in your message.

**severity {get; set;}**

`ccrz.cc_bean_Message.MessageSeverity` that specifies the importance of the message, which applies a default HTML class (which activates corresponding default CSS) to the element where the message displays. The following enumerated values are available:

- SUCCESS
- INFO
- WARN
- ERROR
- CRITICAL

**type {get; set;}**

`ccrz.cc_bean_Message.MessageType` that specifies how the storefront handles the message. The following enumerated values are available:

Value	Usage
ALERT (default)	Show the message in a JavaScript alert window.
STANDARD	Show the message in the <code>&lt;div class="default_page_message"/&gt;</code> HTML element, which renders on storefront pages.
CUSTOM	Show the message in an HTML element whose <code>class</code> attribute value you specify with the <code>displayClass</code> field. This type applies to most use cases for messages in custom subscriber code, and requires that you write front-end code to target a specific element.

## Constructors

This class defines the following constructors:

**ccrz.cc\_bean\_Message**

Default constructor that doesn't take any parameters and creates an empty message with the following field values by default:

Field	Value
<code>type</code>	STANDARD
<code>severity</code>	INFO

**ccrz.cc\_bean\_Message(String message)**

Constructor that takes a message and sets the following field values by default:

Field	Value
type	STANDARD
severity	INFO

**ccrz.cc\_bean\_Message(String message, ccrz.cc\_bean\_Message.MessageSeverity severity)**

Constructor that takes a message and severity, and sets `type` to `STANDARD` by default.

**ccrz.cc\_bean\_Message(String message, ccrz.cc\_bean\_Message.MessageType type)**

Constructor that takes a message and type, and sets `severity` to `INFO` by default.

## Methods

This class doesn't define any methods.

## ccrz.cc\_bean\_MockContactAddress

Represents a `ccrz__E_ContactAddr__c` record.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Properties

This class defines the following Apex [global](#) properties. For more information about Apex properties and accessors, see [Apex Properties](#).

**address1 {get; set;}**

String that specifies the first line of the street address.

**address2 {get; set;}**

String that specifies the second line of the street address.

**address3 {get; set;}**

String that specifies the third line of the street address.

**addrReadOnly {get; set;}**

Boolean that specifies whether the address is read-only.

 **Note:** If this value is null on the source `ccrz__E_ContactAddr__c` record, the bean sets this value to `false`.

**city {get; set;}**

String that specifies the city for the address.

**companyName {get; set;}**

String that specifies the name of a company associated with this address.

**country {get; set;}**

String that specifies the name of the country or territory for the address.

**countryCode {get; set;}**

String that specifies an ISO 3166-1 code for the country or territory.

**dayTime {get; set;}**

String that specifies the primary phone number for the address.

**email {get; set;}**

String that specifies an email associated with the address.

**firstName {get; set;}**

String that specifies the first name of the person associated with the address.

**homePhone {get; set;}**

String that specifies an alternative phone number for the address.

**lastName {get; set;}**

String that specifies the last name of the person associated with the address.

**mailStop {get; set;}**

Unused. This property doesn't control any functionality by default.

**middleName {get; set;}**

String that specifies the middle name of the person associated with the address.

**partnerId {get; set;}**

Unused. This property doesn't control any functionality by default.

**postalCode {get; set;}**

String that specifies the postal code for the address.

**sfdcName {get; set;}**

String that specifies a unique, automatically assigned ID for the record.

**shippingComments {get; set;}**

String that specifies optional shipping comments entered during checkout.

**state {get; set;}**

String that specifies the name of the state for an address that's located *outside* of the United States or Canada.

**stateCode {get; set;}**

String that specifies an ISO 3166-2 code for a US state or Canadian province.

**sfid {get; set;}**

ID of this `ccrz.cc_bean_MockContactAddress` instance.

## Constructors

This class defines the following constructors:

**ccrz.cc\_bean\_MockContactAddress**

Default constructor that doesn't take any parameters and doesn't set any values. When you use this constructor, make sure that you set the individual fields on bean.

**ccrz.cc\_bean\_MockContactAddress(ccrz\_\_E\_ContactAddr\_\_c contactAddress)**

Constructor that sets fields on the `ccrz.cc_bean_MockContactAddress` from the passed-in contact address record.

SEE ALSO:

[ccrz\\_\\_E\\_ContactAddr\\_\\_c](#)

## ccrz.cc\_bean\_MockOrder

Represents a `ccrz__E_Order__c` record.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Properties

This class defines the following Apex [global](#) properties. For more information about Apex properties and accessors, see [Apex Properties](#).

**billTo {get; set;}**

[ccrz.cc\\_bean\\_MockContactAddress](#) that represents the billing address associated with the order.

**buyerEmail {get; set;}**

String that specifies the buyer's email address.

**buyerFirstName {get; set;}**

String that specifies the buyer's first name.

**buyerLastName {get; set;}**

String that specifies the buyer's last name.

**buyerPhone {get; set;}**

String that specifies the buyer's phone number.

**canAmend {get; set;}**

Boolean

Value	Usage
<i>true</i>	The buyer can cancel the order and create another cart with the same items and quantities as the canceled order.
<i>false</i>	The buyer can't cancel the order and create another cart.

**canCancel {get; set;}**

Boolean

Value	Usage
<i>true</i>	The buyer can cancel the order.
<i>false</i>	The buyer can't cancel the order.

**canReorder {get; set;}**

Boolean

Value	Usage
<i>true</i>	The buyer can create another cart with the same items and quantities as the canceled order.
<i>false</i>	The buyer can't create another cart with the same items and quantities as the canceled order.

**cartId {get; set;}**

String that specifies the ID of the cart used for placing the order.

**currencyIsoCode {get; set;}**String that specifies the ISO 4217 currency code for the order, such as *USD* or *JPY*.**encryptedId {get; set;}**

String that specifies a semi-random UUID to refer to the order in page URLs.

**externalOrderId {get; set;}**

String that specifies a unique ID that associates the record with an external system.

**extrinsic {get; set;}**

Map&lt;String, Object&gt; that represents a placeholder for external data that you can pass to this bean and other objects that refer to this bean.

**isLL {get; set;}**

Boolean

Value	Usage
<i>true</i>	The buyer placed the order using line-level independence (LLI) checkout.
<i>false</i>	The buyer placed the order using standard checkout.

**name {get; set;}**

String that specifies the order's auto-numbered name.

**orderAccountData {get; set;}**

Map<String, Object> that describes the account used for placing the order.

**orderDate {get; set;}**

Date that specifies the date when the buyer placed the order.

**orderDateStr {get; set;}**

String representation of orderDate.

**orderItemGroups {get; set;}**

List<ccrz\_\_E\_OrderItemGroup\_\_c> that describes shipping groups for the order when the buyer used LLI checkout.

**orderNumber {get; set;}**

Decimal that specifies the digit portion of the order's name auto-number value, plus 1,000,000. For example, for an order with an auto-numbered name of 0-0000000018, the orderNumber is 1,000,018.

**orderNumberAsString {get; set;}**

String representation of orderNumber.

**orderShipAmount {get;}**

String representation of shipAmount.

**orderSubTotal {get;}**

String representation of subTotalAmount.

**orderTaxAmount {get;}**

String representation of taxAmount.

**orderTotalSurcharge {get;}**

String representation of totalSurcharge.

**ownerName {get; set;}**

String that specifies the user that owns the order record.

**payments {get; set;}**

List<ccrz.cc\_bean\_Payment> that specifies payments applied to the cart.

**requestDate {get; set;}**

String that specifies the delivery date that the buyer requested during checkout.

**requestDateVal {get; set;}**

Date representation of requestDate.

**sfid {get; set;}**

Salesforce 18-digit ID of the order record.

**shipAmount {get; set;}**

Double that specifies the amount calculated for shipping the order during checkout.

**shipComplete {get; set;}**

Boolean

Value	Usage
<i>true</i>	The buyer selected the Ship Complete option during checkout.
<i>false</i>	The buyer didn't select the Ship Complete option during checkout.

**shippingMethod {get; set;}**

String that specifies the order's selected shipping method.

**shipTo {get; set;}**

[ccrz.cc\\_bean\\_MockContactAddress](#) that represents the shipping address associated with the order.

**shipTrackNo {get; set;}**

String that specifies a tracking number for the order's shipment.

**status {get; set;}**

String that specifies the status of the order shown in the My Orders section of the My Account page.

**subTotalAmount {get; set;}**

Double that specifies the total entitled price of all items in the order.

**taxAmount {get; set;}**

Double that specifies the tax amount calculated during checkout.

**totalAmount {get; set;}**

Double that specifies the total cost calculated for the order.

**totalSurcharge {get; set;}**

Double that specifies charges other than taxes or shipping applied to the cart.

**transactionPayments {get; set;}**

List<Map<String, Object>> that describes [ccrz\\_\\_E\\_TransactionPayment\\_\\_c](#) records applied to the cart.

## Constructors

This class defines the following constructors:

**ccrz.cc\_bean\_MockOrder**

Default constructor that doesn't take any parameters and doesn't set any values. When you use this constructor, make sure that you set the individual fields on bean.

**ccrz.cc\_bean\_MockOrder(ccrz\_\_E\_Order\_\_c order)**

Constructor that sets fields on the [ccrz.cc\\_bean\\_MockOrder](#) from the passed-in order record.

**ccrz.cc\_bean\_MockOrder(ccrz.cc\_bean\_MyOrderItem order)**

Constructor that sets fields on the [ccrz.cc\\_bean\\_MockOrder](#) from the passed-in [ccrz.cc\\_bean\\_MyOrderItem](#).

## Methods

This class doesn't define any methods.

SEE ALSO:

[ccrz\\_\\_E\\_Order\\_\\_c](#)


[Salesforce Help Allow a Buyer to Manage Their Orders on the My Account Page](#)

## ccrz.cc\_bean\_PromoRD

Passes promotion details between other global API method calls and returns JSON data for showing promotions on the storefront.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.cc\\_bean\\_PromoRD](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## ccrz.cc\_ctrl\_hlpr\_ShippingOption

Represents a shipping option available in the Shipping Method picklist on the storefront's Checkout page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Properties

This class defines the following Apex `global` properties. For more information about Apex properties and accessors, see [Apex Properties](#).

**currencyCode {get; set;}**

String that specifies the ISO 4217 currency code, such as `USD` or `JPY`, for the shipping option.

**discount {get; set;}**

Decimal that specifies the amount of [discount on the shipping cost](#).

**discountedShipCost {get; set;}**

Decimal that specifies the cost of the shipping option after applying an applicable discount on the shipping cost.

**price {get; set;}**

Decimal that specifies the total cost of the shipping option.

**provider {get; set;}**

String that specifies the shipping service provider.

**serviceName {get; set;}**


String that specifies the service name.

**systemReference {get; set;}**

String that specifies a reference to an external system.

**uniqueId {get; set;}**

String that specifies a unique ID for the shipping option. By default, this ID is created by appending `provider` to `serviceName`.

 **Note:** This value is mandatory for persisting the ship method to a placed order.



## Constructors

This class defines the following constructors:

### `ccrz.cc_ctrl_hlpr_ShippingOption`

Default constructor that doesn't take any parameters and doesn't set any fields.

### `ccrz.cc_ctrl_hlpr_ShippingOption(String provider, String serviceName, Decimal price)`

Constructor that takes and sets a provider, service name, and price. This constructor also sets `uniqueId`.

### `ccrz.cc_ctrl_hlpr_ShippingOption(String provider, String serviceName, Decimal price, Decimal discountedShipCost)`

Constructor that takes and sets a provider, service name, price, and discounted shipping cost. This constructor also sets `uniqueId`.

### `ccrz.cc_ctrl_hlpr_ShippingOption(String provider, String systemReference, String serviceName, Decimal price, Decimal discountedShipCost)`

Constructor that takes and sets a provider, external system reference, service name, price, and discounted shipping cost. This constructor also sets `currencyCode` to `null`.

### `ccrz.cc_ctrl_hlpr_ShippingOption(String provider, String systemReference, String serviceName, String currencyCode, Decimal price, Decimal discountedShipCost)`

Constructor that takes and sets a provider, external system reference, service name, price, currency code, and discounted shipping cost.

## Methods

This class defines the following methods:

### `getNetPrice()`

Returns a `Decimal` that results from subtracting the shipping discount (`discount`) from the original shipping cost (`price`).

SEE ALSO:

[ccrz\\_\\_E\\_ShippingRate\\_\\_c](#)

[ccrz.cc\\_api\\_ShippingAndHandling](#)

[Salesforce Help Specify Shipping Options for Standard Checkout](#)

[Salesforce Help Offer a Discount on the Total Shipping Cost](#)

## ccrz.cc\_CallContext

Provides a standard interface for referencing the current user, storefront activity, and other session information required for B2B Commerce for Visualforce API calls.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Initialization

If you're calling the API from a Visualforce page, the context is already created, and you can reference the context's static fields.

If you're calling the API from an Apex remote action, you must initialize the context using `ccrz.cc_RemoteActionContext` before you can reference its fields.

## Properties

This class defines the following Apex `global` properties. For more information about Apex properties and accessors, see [Apex Properties](#).

### **basePriceList {get; set;}**

String that specifies the [storefront's base price list](#).

### **currAccount {get; set;}**

[Account](#) associated with the current user. The following fields on the account record are available:

- Id (18 digits)
- AccountNumber
- Name
- ParentId
- `ccrz__E_AccountGroup__r.Id`
- `ccrz__TaxExemptAccount__c`

### **currAccountGroup {get; set;}**

`ccrz__E_AccountGroup__c` that the current user's account belongs to.

### **currAccountId {get; set;}**

String that specifies the 18-digit Salesforce ID of the account associated with the current user. For a guest user, this ID is for the account named Anonymous.

### **currCartId {get; set;}**

String that specifies the encrypted ID of the current cart, if available.

### **currContact {get; set;}**

[Contact](#) associated with the current user. The following fields on the contact record are available:

- Id
- Email
- FirstName
- LastName
- Phone

### **currPageKey {get; set;}**

String that specifies the `currPageName` value. If the current storefront page is a subscriber page, this value is your subscriber page's key.

**currPageName {get; set;}**

String that specifies the current storefront page without the namespace prefix, such as `HomePage` instead of `ccrz__HomePage`.

**currPageParameters {get; set;}**

Map<String, String> of the current page's URL parameters.

- If the context is within a Visualforce page, the parameters are returned from the [ApexPages.currentPage.getParameters](#) method.
- If the context is within an Apex remote action, the parameters are from the page where the remote action was invoked. If you're using a URL rewriter, these parameters are the original page parameters, not any rewritten parameters.

**currPageUIKey {get; set;}**

String that specifies the `currUIKey` value. If the current storefront page is a subscriber page, this value is your subscriber page's key.

**currUIKey {get; set;}**

String that specifies the page key for the current storefront page, such as `HP` for the Home page.

**currURL {get; set;}**

String that specifies the URL of the current storefront page.

- If the context is within a Visualforce page, the URL is returned from the [ApexPages.currentPage.getUrl](#) method.
- If the context is within an Apex remote action, the URL is from the page where the remote action was invoked.

**currUser {get; set;}**

User record for the current user.



**Note:** If a [customer service representative \(CSR\)](#) is logged in on behalf of a storefront user, this user is the storefront user, not the CSR user.

The following fields on the user record are available:

- Id
- AccountId
- Contact Id
- Email
- LanguageLocaleKey
- LocaleSidKey
- ProfileId
- Username
- ccrz\_\_CC\_CurrencyCode\_\_c
- ccrz\_\_TaxExempt\_\_c

**currUserId {get; set;}**

String that specifies the ID of the current user.



**Note:** If a CSR is logged in on behalf of a storefront user, this ID is the storefront user's ID, not the CSR user's ID.

**effAccount {get; set;}**

Account associated with a currently selected [Allow a Buyer to Purchase for Multiple Effective Accounts](#).

- If the current buyer is a guest, this value is null.
- If the current buyer is a registered user but isn't using an effective account, this value is `currAccount`.

The following fields on the account record are available:

- Id
- AccountNumber
- Name
- ParentId
- ccrz\_\_E\_AccountGroup\_\_r.Id
- ccrz\_\_TaxExemptAccount\_\_c

**effAccountGroup {get; set;}**

`ccrz__E_AccountGroup__c` associated with the currently selected [effective account](#).

- If the current buyer is a guest, a request for this field returns an exception.
- If the current buyer is a registered user but isn't using an effective account, this value is `currAccountGroup`.

**effAccountId {get; set;}**

String that specifies the account ID of the currently selected [effective account](#).

- If the current buyer is a guest, this value is an empty string.
- If the current buyer is a registered user but isn't using an effective account, this value is `currUserId`.

**entAccountId {get; set;}**

String that specifies the ID of the account to use for product entitlement.

- If the current user is a guest user, this value is the same as `currAccountId`.
- Otherwise, this value is the same as `effAccountId`.

**entAccount {get; set;}**

[Account](#) to use for product entitlement.

- If the current user is a guest user, this value is the same as `currAccount`.
- Otherwise, this value is the same as `effAccount`.

**entAccountGroup {get; set;}**

`ccrz__E_AccountGroup__c` associated with `entAccount`.

**isGuest {get; set;}**

Boolean

Value	Usage
<i>true</i>	The current buyer is a guest.
<i>false</i>	The current buyer is a registered user.

**isOnBehalf {get; set;}**

Boolean

Value	Usage
<i>true</i>	The user that's currently logged in is different than the user that the <code>portalUser</code> URL parameter specifies, such as when a CSR is logged in.
<i>false</i>	The <code>portalUser</code> URL parameter specifies the user that's currently logged in.

**isRemoteCall {get; set;}**

Boolean

Value	Usage
<i>true</i>	The context is within an Apex remote action, or some other external source.
<i>false</i>	The context is within a Visualforce page.

**priceGroup {get; set;}**

`ccrz__E_PriceGroup__c` associated with the current context's account. The [grid URL parameter](#) specifies this price group.

**priceGroupId {get; set;}**

String that specifies the ID of the price group associated with the current context's account. This ID matches the value of the `grid` URL parameter.

**remoteContext**

`ccrz.cc_RemoteActionContext` instance when the context is within an Apex remote action.

**storefront {get; set;}**

String that specifies the current storefront name.

**storefrontSettings {get; set;}**

Map<[String](#), [Object](#)> that describes the current [storefront configuration settings](#).

**userCurrency {get; set;}**

String that specifies the current user's currency.

**userLocale {get; set;}**

String that specifies the current user's [locale](#).

## Methods

### IN THIS SECTION:

#### [init](#)

Wraps a call to the `initRemoteContext` method, which initializes the `ccrz.cc_CallContext` within a passed remote context, and returns a `ccrz.cc_RemoteActionResult`.

#### [initRemoteContext](#)

Initializes the `ccrz.cc_CallContext` within an Apex remote action.

#### [isConfigEqual](#)

Evaluates whether the current page has a storefront configuration setting with a specific value. This method only considers configuration settings from the most recently activated configuration cache.

#### [isConfigTrue](#)

Evaluates whether the current page has a storefront configuration setting with the value `TRUE`. This method only considers configuration settings from the most recently activated configuration cache.

[getConfigValue](#)

Return the current value for a particular storefront configuration setting. This method only considers configuration settings from the most recently activated configuration cache.

SEE ALSO:

[ccrz.cc\\_RemoteActionContext](#)

[Salesforce Help Allow a Buyer to Purchase for Multiple Effective Accounts](#)

## ccrz.cc\_CallContext.init

Wraps a call to the `initRemoteContext` method, which initializes the `ccrz.cc_CallContext` within a passed remote context, and returns a `ccrz.cc_RemoteActionResult`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Important:** Call this method only from an Apex remote action, and not from a Visualforce page.

## Signature

```
global static ccrz.cc_RemoteActionResult init(ccrz.cc_RemoteActionContext)
```

## Inputs

### [ccrz.cc\\_RemoteActionContext](#)

The class that instantiates a new context from an Apex remote action.

## Outputs

### [ccrz.cc\\_RemoteActionResult](#)

Initially, the result's `success` is `false`. Make sure that you set `success` to `true` before you return data from the remote action.

### Example: **Example**

```
@RemoteAction
global static ccrz.cc_RemoteActionResult customRemoteActionMethod(final
```

```

ccrz.cc_RemoteActionContext ctx, other parameters) {
    ccrz.cc_RemoteActionResult result = ccrz.cc_CallContext.init(ctx);

    try {
        // Your custom business logic
        result.data = result of business logic; //data to return
        result.success = true;
    } catch (Exception e) {
        // Error handling, perhaps using result.messages to return error data
    }
    return res;
}

```

SEE ALSO:

[ccrz.cc\\_RemoteActionContext](#)

[ccrz.cc\\_RemoteActionResult](#)

## ccrz.cc\_CallContext.initRemoteContext

Initializes the `ccrz.cc_CallContext` within an Apex remote action.



**Important:** Call this method only from an Apex remote action, and not from a Visualforce page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static void initRemoteContext(ccrz.cc_RemoteActionContext)
```

## Inputs

[ccrz.cc\\_RemoteActionContext](#)

The class that instantiates a new context from an Apex remote action.

## Outputs

This method doesn't return any data.

### Example: Example

```
@RemoteAction
global static void customRemoteActionMethod {
    ccrz.cc_RemoteActionContext ctx = new ccrz.cc_RemoteActionContext;
    ctx.storefront = 'DefaultStore';
    ccrz.cc_CallContext.initRemoteContext (ctx);
}
```

SEE ALSO:

[ccrz.cc\\_RemoteActionContext](#)

## ccrz.cc\_CallContext.isConfigEqual

Evaluates whether the current page has a storefront configuration setting with a specific value. This method only considers configuration settings from the most recently activated configuration cache.

## Compatibility


This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Boolean isConfigEqual (String, String, String)
```

## Inputs

 **Tip:** This method transforms any string values to lowercase. As a best practice, always use lowercase strings when evaluating configuration settings.

### **config**

String that specifies the configuration setting API name to evaluate, such as *pl.defsort* for the Default Sort configuration of the Product List module.

### **value**

String that specifies the configuration value to evaluate, such as *price*.



**pageUIKey**

A specific page key to evaluate, such as `p1p` for Product List page.

## Outputs

Boolean

Value	Usage
true	The specified configuration has a case-insensitive variation of the value to evaluate, such as <code>PRICE</code> , <code>Price</code> , or <code>price</code> .
false	The specified configuration doesn't exist or doesn't contain a case-insensitive variation of the value to evaluate.

SEE ALSO:

[Salesforce Help Manage Configuration Settings](#)

## ccrz.cc\_CallContext.isConfigTrue

Evaluates whether the current page has a storefront configuration setting with the value `TRUE`. This method only considers configuration settings from the most recently activated configuration cache.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Boolean isConfigTrue(String, String)
```

## Inputs



**Tip:** This method transforms any string values to lowercase. As a best practice, always use lowercase strings when evaluating configuration settings.

**config**

String that specifies the configuration setting API name to evaluate, such as `ct.enabled` for the Enabled configuration of the Category Tree module.

**pageUIKey**

A specific page key to evaluate, such as `hp` for Home page.

## Outputs

Boolean

Value	Usage
<code>true</code>	The specified configuration has a case-insensitive variation of <code>TRUE</code> , including <code>true</code> or <code>True</code> .
<code>false</code>	The specified configuration doesn't exist or doesn't contain a case-insensitive variation of <code>TRUE</code> .

SEE ALSO:

[Salesforce Help Manage Configuration Settings](#)

## ccrz.cc\_CallContext.getConfigValue

Return the current value for a particular storefront configuration setting. This method only considers configuration settings from the most recently activated configuration cache.

## Compatibility


This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static String getConfigValue(String, String)
```

## Inputs

 **Tip:** This method transforms any string values to lowercase. As a best practice, always use lowercase strings when evaluating configuration settings.

**config**

String that specifies the configuration setting API name to evaluate, such as `ct.enabled` for the Enabled configuration of the Category Tree module.

**pageUIKey**

A specific page key to evaluate, such as `hp` for Home page.

## Outputs

String of the specified configuration setting's value. If the specified configuration doesn't exist, this value is null.

SEE ALSO:

[Salesforce Help Manage Configuration Settings](#)

## ccrz.cc\_RemoteActionContext

Initializes `ccrz.cc_CallContext` from an Apex remote action.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Properties

This class defines the following Apex [global](#) properties. For more information about Apex properties and accessors, see [Apex Properties](#).

**currentCartId {get; set;}**

String that specifies the encrypted ID of the current cart, if available.

**currentPageName {get; set;}**

String that specifies the current storefront page without the namespace prefix, such as `HomePage` instead of `ccrz__HomePage`. This field is used with methods of `ccrz.cc_CallContext` to determine whether any storefront configuration settings exist for the specific page.

**currentPageURL {get; set;}**

String that specifies the URL for the current storefront page.

**effAccountId {get; set;}**

String that specifies the account ID of the currently selected [effective account](#).

- If the current buyer is a guest, this value is an empty string.

- If the current buyer is a registered user but isn't using an effective account, this value is `currUserId`.

**portalUserId {get; set;}**

String that specifies the ID of the storefront user that a [customer service representative \(CSR\) logs in on behalf of](#).

**priceGroupId {get; set;}**

String that specifies the ID of the price group associated with the current context's account. This ID matches the value of the `grid` URL parameter.

**queryParams {get; set;}**

Map<String, String> that specifies the [URL parameters](#) for the current storefront page at the time of the Apex remote action call.

**storefront {get; set;}**

String that specifies the current storefront name.

**userIsoCode {get; set;}**

String that specifies the currency for the current context. If this remote action context queries cart data, match the cart's currency, or leave this field blank.

**userLocale {get; set;}**

String that specifies the locale for the current context. Any subsequent API call returns data that applies only to this locale.

## Constructors

This class defines a single constructor, `ccrz.cc_RemoteActionContext`, which doesn't take any parameters.

### Example: Example

```
@RemoteAction
    global static ccrz.cc_RemoteActionResult customRemoteActionMethod(final
ccrz.cc_RemoteActionContext ctx, other parameters) {
    ccrz.cc_RemoteActionResult result = ccrz.cc_CallContext.init(ctx);

    try {
        // Your custom business logic
        result.data = result of business logic; //data to return
        result.success = true;
    } catch (Exception e) {
        // Error handling, perhaps using result.messages to return error data
    }
    return res;
}
```

SEE ALSO:

[ccrz.cc\\_CallContext](#)

## ccrz.cc\_RemoteActionResult

Returns the result of an Apex remote action.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Properties

This class defines the following Apex `global` properties. For more information about Apex properties and accessors, see [Apex Properties](#).

### **data** {get; set;}

Object that represents the result of the remote action to return to the user interface. You can serialize the object as JSON data.

### **inputContext** {get; set;}

`ccrz.cc_RemoteActionContext` for initializing `ccrz.cc_CallContext` within the remote action.



**Note:** This value is just an echo of the data passed in, and doesn't define any other functionality.

### **messages** {get; set;}

`List<ccrz.cc_bean_Message>` that contains error messages to return to the user interface. This list is `final` within the class.

### **success** {get; set;}

Boolean

Value	Usage
<code>true</code>	The remote action completed successfully.
<code>false</code>	The remote action encountered errors.

## Constructors

This class defines a single constructor, `ccrz.cc_RemoteActionResult`, which doesn't take any parameters.

The constructor creates a new `ccrz.cc_RemoteActionResult` with the following fields:

- `success` is `true`
- `data` is `false`
- `inputContext` is `ccrz.cc_RemoteActionContext`

### Example: Example

```
@RemoteAction
global static ccrz.cc_RemoteActionResult customRemoteActionMethod(final
```

```
ccrz.cc_RemoteActionContext ctx, other parameters) {  
    ccrz.cc_RemoteActionResult result = ccrz.cc_CallContext.init(ctx);  
  
    try {  
        // Your custom business logic  
        result.data = result of business logic; //data to return  
        result.success = true;  
    } catch (Exception e) {  
        // Error handling, perhaps using result.messages to return error data  
    }  
    return res;  
}
```

SEE ALSO:


[ccrz.cc\\_CallContext](#)

## ccrz.ccMultiListIterator

Iterates through multiple Apex lists and creates `List<Object>` representations. This class is used with batch jobs where the batched data can't be constructed with SOQL.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccMultiListIterator](#)


 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## ccrz.ccMultiSetIterator

Iterates through multiple Apex sets and creates `List<Object>` representations. This class is used with batch jobs where the batched data can't be constructed with SOQL.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccMultiSetIterator](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.


## ccrz.ccPaymentProcessor

Provides an extension point class for processing a specific payment type for all payment flows, including checkout, subscriptions, and invoices.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Important:** The information in these help topics applies only to payment processing during checkout. For information about processing payments for invoices and subscriptions, see the PDF archived from CloudCraze documentation:

(PDF) [ccrz.ccPaymentProcessor](#)

## Methods

### IN THIS SECTION:

#### [preProcess](#)

Executes the first step of any payment flow and performs Apex callouts required for processing the payment. This method supports Apex callouts.

#### [process](#)

Executes the transactional step of a payment flow. This method does *not* support Apex callouts.

#### [postProcess](#)

Handles the success or failure result of the `ccrz.ccPaymentProcessor.process` transaction. This method does *not* support Apex callouts.


## ccrz.ccPaymentProcessor.preProcess

Executes the first step of any payment flow and performs Apex callouts required for processing the payment. This method supports Apex callouts.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

 **Warning:** If you override this method in custom subscriber code, avoid any DML operations.

## Signature

```
global virtual Map<String, Object> preprocess (Map<String, Object>)
```

## Inputs for Checkout

The `ccrz.cc_hk_Payment.processPayment` method invokes this method when processing a payment during checkout. This method accepts a `Map<String, Object>` that contains the following keys:

### `ccrz.cc_hk_Payment.PARAM_ACCOUNT_TYPE`

String, based on the input data's `accountType`, that specifies the transaction's method of payment, such as purchase order (PO), credit card, and so on.

### `ccrz.cc_hk_Payment.PARAM_CART`

`ccrz__E_Cart__c` record that's being converted to an order.

### `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX`

String that specifies the context of the payment. For checkout payments, this value is `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_CHECKOUT`.

### `ccrz.cc_hk_Payment.PARAM_STORED_PAYMENT`

`ccrz__E_StoredPayment__c` that represents an original stored payment used for the transaction payment details.

### `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA`

JSON-formatted string of transaction payment properties and their values passed into `ccrz.cc_hk_Payment.processPayment`.

### `ccrz.cc_hk_Payment.PARAM_TRANSACTION_PROCESSED_DATA`

`ccrz.cc_hk_Payment.TransactionPaymentParams` object that represents the original input `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA`, updated with the cart's `amount` set.

## Inputs for Invoices or Subscriptions

For information about processing payments for invoices and subscriptions, see the PDF archived from CloudCraze documentation: (PDF) [ccrz.ccPaymentProcessor](#)

## Outputs

`Map<String, Object>` that includes all input keys.

SEE ALSO:

[ccrz.cc\\_hk\\_Payment.processPayment](#)

[ccrz\\_\\_E\\_StoredPayment\\_\\_c](#)

[Invoking Callouts Using Apex](#)

## `ccrz.ccPaymentProcessor.process`

Executes the transactional step of a payment flow. This method does *not* support Apex callouts.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> process (Map<String, Object>)
```

## Inputs for Checkout

During the checkout process, this method is invoked as the final step of the transaction when placing an order. This method accepts a `Map<String, Object>` that contains the following keys:

### **ccrz.cc\_hk\_Payment.PARAM\_ACCOUNT\_TYPE**

String, based on the input data's `accountType`, that specifies the transaction's method of payment, such as purchase order (PO), credit card, and so on.

### **ccrz.cc\_hk\_Payment.PARAM\_ORDER\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Order__c` record created from the cart.

### **ccrz.cc\_hk\_Payment.PARAM\_PAYMENT\_CTX**

String that specifies the context of the payment. For checkout payments, this value is `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_CHECKOUT`.

### **ccrz.cc\_hk\_Payment.PARAM\_PRIMARY\_TRANSACTION\_RESULTS**

`Map<String, Object>` that contains the results from the `ccrz.cc_hk_Order.createTransaction` method.

### **ccrz.cc\_hk\_Payment.PARAM\_SUBS\_RESULTS**

`Map<String, Object>` that contains the results from the `ccrz.cc_hk_Subscriptions.onOrder` method.

### **ccrz.cc\_hk\_Payment.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed into `ccrz.cc_hk_Payment.processPayment`.

### **ccrz.cc\_hk\_Payment.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object that represents the original input `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA`, updated with the cart's `amount` set.

## Inputs for Invoices or Subscriptions

For information about processing payments for invoices and subscriptions, see the PDF archived from CloudCraze documentation:

(PDF) [ccrz.ccPaymentProcessor](#)

## Outputs

Map<String, Object> that includes all input keys.

SEE ALSO:

[ccrz.cc\\_hk\\_Payment.processPayment](#)

[ccrz.cc\\_hk\\_Order.createTransaction](#)

## ccrz.ccPaymentProcessor.postProcess

Handles the success or failure result of the `ccrz.ccPaymentProcessor.process` transaction. This method does *not* support Apex callouts.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global virtual Map<String, Object> postProcess (Map<String, Object>)
```

## Inputs for Successful Checkout

If the `ccrz.ccPaymentProcessor.process` method completed successfully, this method accepts a Map<String, Object> that contains the following keys for evaluating the successful transaction:

### **ccrz.cc\_hk\_Payment.PARAM\_ACCOUNT\_TYPE**

String, based on the input data's `accountType`, that specifies the transaction's method of payment, such as purchase order (PO), credit card, and so on.

### **ccrz.cc\_hk\_Payment.PARAM\_ORDER\_ID**

String that specifies the Salesforce ID of the `ccrz__E_Order__c` record created from the cart.

### **ccrz.cc\_hk\_Payment.PARAM\_PAYMENT\_CTX**

String that specifies the context of the payment. For checkout payments, this value is `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_CHECKOUT`.

### **ccrz.cc\_hk\_Payment.PARAM\_PRIMARY\_TRANSACTION\_RESULTS**

Map<String, Object> that contains the results from the `ccrz.cc_hk_Order.createTransaction` method.

**ccrz.cc\_hk\_Payment.PARAM\_SUBS\_RESULTS**

Map<String, Object> that contains the results from the `ccrz.cc_hk_Subscriptions.onOrder` method.

**ccrz.cc\_hk\_Payment.PARAM\_SUCCESS**

Boolean that must be `true`.

**ccrz.cc\_hk\_Payment.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed into `ccrz.cc_hk_Payment.processPayment`.

**ccrz.cc\_hk\_Payment.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object that represents the original input `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA`, updated with the cart's `amount` set.

## Inputs for Failed Checkout

If the `ccrz.ccPaymentProcessor.process` method returned an exception, this method accepts a Map<String, Object> that contains the following keys for evaluating the failed transaction:

**ccrz.cc\_hk\_Payment.PARAM\_CART**

`ccrz__E_Cart__c` record that's being converted to an order.

**ccrz.cc\_hk\_Payment.PARAM\_ERROR**

The exception that `ccrz.ccPaymentProcessor.process` returned.

**ccrz.cc\_hk\_Payment.PARAM\_PAYMENT\_CTX**

String that specifies the context of the payment. For checkout payments, this value is `ccrz.cc_hk_Payment.PARAM_PAYMENT_CTX_CHECKOUT`.

**ccrz.cc\_hk\_Payment.PARAM\_SUCCESS**

Boolean that must be `false`.

**ccrz.cc\_hk\_Payment.PARAM\_TRANSACTION\_DATA**

JSON-formatted string of transaction payment properties and their values passed into `ccrz.cc_hk_Payment.processPayment`.

**ccrz.cc\_hk\_Payment.PARAM\_TRANSACTION\_PROCESSED\_DATA**

`ccrz.cc_hk_Payment.TransactionPaymentParams` object that represents the original input `ccrz.cc_hk_Payment.PARAM_TRANSACTION_DATA`, updated with the cart's `amount` set.

## Inputs for Invoices or Subscriptions

For information about processing payments for invoices and subscriptions, see the PDF archived from CloudCraze documentation:

(PDF) [ccrz.ccPaymentProcessor](#)

## Outputs

Map<String, Object> that includes all input keys.

SEE ALSO:

[ccrz.cc\\_hk\\_Payment.processPayment](#)

[ccrz.cc\\_hk\\_Order.createTransaction](#)

## ccrz.ccProductIndexBuilder

Builds product index records by executing a series of synchronously executed Apex batch jobs. Each batch job includes a `finish` method that calls back into the `ccrz.ccProductIndexBuilder` class for starting the next batch job.

### Compatibility

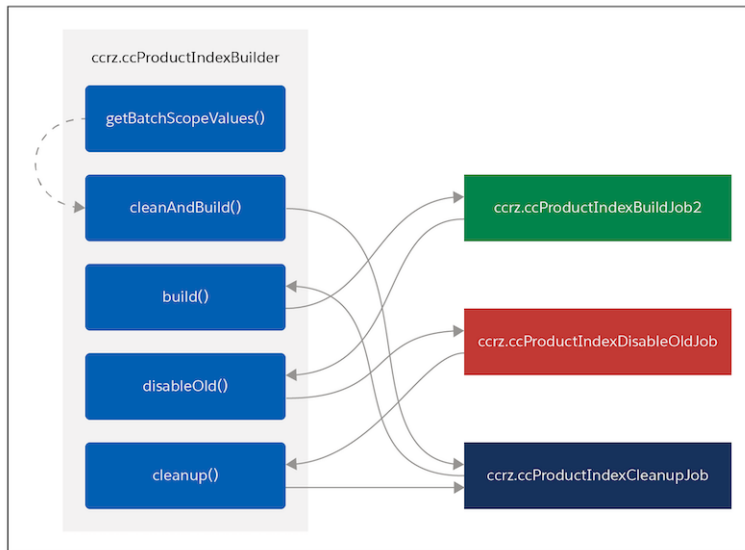
This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Execution Order

The methods of the `ccrz.ccProductIndexBuilder` class execute when you [refresh product index records from CC Admin](#).

The methods of this class and the related batch job classes execute in the following order.



Your version of B2B Commerce determines where the `ccrz.ccProductIndexBuilder` execution begins.

B2B Commerce Managed Package Version	Method Where Execution Begins
Spring '20 (version 4.12)	<code>getBachScopeValues</code>

B2B Commerce Managed Package Version	Method Where Execution Begins
Summer '19 (version 4.11) or Spring '19 (version 4.10)	cleanAndBuild
Summer '18 (version 4.9)	build

## Methods

### IN THIS SECTION:

#### [getBatchScopeValues](#)

Divides all the org's product records into groups for building product index records more efficiently.

#### [cleanAndBuild](#)

Executes the `ccrz.ccProductIndexCleanupJob` batch job before the next build process begins. The `ccrz.ccProductIndexCleanupJob` batch job removes existing product index records where the Active field is `FALSE`.

#### [build](#)

Executes the `ccrz.ccProductIndexBuildJob2` batch job for building product index records.

#### [disableOld](#)

Executes the `ccrz.ccProductIndexDisableOldJob` batch job, which updates the Active field on existing product index records. All product index records where Active is `FALSE` update so that Active is `TRUE`, and vice versa.

#### [cleanup](#)

Executes the `ccrz.ccProductIndexCleanupJob` batch job, which removes existing product index records where the Active field is `FALSE`.

### SEE ALSO:

[Salesforce Help Maintain Product Index Records](#)

## ccrz.ccProductIndexBuilder.getBatchScopeValues

Divides all the org's product records into groups for building product index records more efficiently.

### Compatibility

This utility method is available *only* in B2B Commerce for Visualforce Spring '20 (managed package version 4.12 and API version 11).

### Signature

```
global static Map<Integer, Map<String, String>> getBatchScopeValues (Map<String, Object>)
```

### Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

**ccrz.ccProductIndexBuilder.IS\_FULL\_REFRESH**  
Boolean

Value	Usage
<i>TRUE</i> (default)	Divide all the org's product records evenly among the batch groups.
<i>FALSE</i>	Evaluate only the products where the <a href="#">Product Index Status is Not Current</a> , and divide those products evenly among the batch groups.

### **ccrz.ccProductIndexBuilder.NUMBER\_OF\_SCOPE\_KEY\_PAIRS**

Integer that specifies how many groups you want to divide the product index build process into. You can specify up to five (5) groups.

 **Note:** When an admin selects how many parallel batch jobs they want to use for refreshing product index records, this key is set for this method.

## Outputs

This method returns a `Map<Integer, Map<String, String>>` of the specified batch groups. Each group defines a `START_KEY` and `END_KEY`, which is the product ID where the batch group begins and ends, respectively.

### Example: Example

Divide the product index build jobs into five groups, and evaluate only the products where Product Index Status is Not Current.

```
ccrz.ccProductIndexBuilder.getBatchScopeValues(new Map<String, Object> {
    ccrz.ccProductIndexBuilder.NUMBER_OF_SCOPE_KEY_PAIRS => 5,
    ccrz.ccProductIndexBuilder.IS_FULL_REFRESH => false
});
```

SEE ALSO:

[Salesforce Help Values for Product Index Status](#)

## ccrz.ccProductIndexBuilder.cleanAndBuild

Executes the `ccrz.ccProductIndexCleanupJob` batch job before the next build process begins. The `ccrz.ccProductIndexCleanupJob` batch job removes existing product index records where the `Active` field is `FALSE`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature


```
global static Map<String, Object> cleanAndBuild (Map<String, Object>)
```

## Inputs (Optional)

`Map<String, Object>` that can include the following keys:

### `ccrz.ccApiProductIndex.CLEAN_ONLY`

Boolean

Value	Usage
<code>TRUE</code>	The next build process updates product index records only for products where Product Index Status is Not Current.   <b>Note:</b> When an admin selects Update Only Outdated Products when refreshing product index records, this key value is set for this method.
<code>FALSE</code> (default)	The next build process updates product index records for all the org's products.

### `ccrz.ccApiProductIndex.LOCALES`

`Set<String>` of specific [storefront locales, including specific and generic locales](#), whose existing product index records you want to clean.

```
new Set<String>{'Locale_1', 'Locale_2'}
```

If you don't specify any locales, the method evaluates all locales across all your org's storefronts.

### `ccrz.ccApiProductIndex.MASTER_PRICE_LIST`

String that specifies the ID of the storefront's [master price list](#), if applicable.

### `ccrz.ccProductIndexBuilder.BATCH_JOB_PRODUCT_SCOPE`

`Map<String, Object>` that specifies a range of product IDs to build product index records for. The `ccrz.ccProductIndexBuilder.getBatchScopeValues` method determines this range. The map's `START_KEY` specifies the beginning of the range, and the `END_KEY` specifies the conclusion of the range.

 **Note:** This key is available *only* in B2B Commerce for Visualforce Spring '20 (version 4.12) or later.

## Outputs

`Map<String, Object>`, which contains the same input data that was passed to `ccrz.ccProductIndexCleanupJob`.

SEE ALSO:

[Salesforce Help Values for Product Index Status](#)

[Salesforce Help Strategies for Managing Price List Item Currencies with Product Index Records](#)

[Salesforce Help Implement the Primary Index for All Currencies Strategy for Product Index Records](#)

## `ccrz.ccProductIndexBuilder.build`

Executes the `ccrz.ccProductIndexBuildJob2` batch job for building product index records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature


```
global static Map<String, Object> build(Map<String, Object>)
```

## Inputs (Optional)

This method accepts the `Map<String, Object>` from `ccrz.ccProductIndexCleanupJob`. The map can specify the following keys:

### `ccrz.ccApiProductIndex.CLEAN_ONLY`

Boolean

Value	Usage
<code>TRUE</code>	Build product index records only for products where Product Index Status is Not Current.   <b>Note:</b> When an admin selects Update Only Outdated Products when refreshing product index records, this key value is set for this method.
<code>FALSE</code> (default)	Build product index records for all the org's products.

### `ccrz.ccApiProductIndex.LOCALES`

`Set<String>` of storefront locales, including specific and generic locales, to build product index records for.

```
new Set<String>{'Locale_1', 'Locale 2'}
```

If you don't specify any locales, the method evaluates all locales across all your org's storefronts.

### `ccrz.ccApiProductIndex.MASTER_PRICE_LIST`

String that specifies the ID of the storefront's [master price list](#), if applicable.

### `ccrz.ccProductIndexBuilder.BATCH_JOB_PRODUCT_SCOPE`

`Map<String, Object>` that specifies a range of product IDs to build product index records for. The `ccrz.ccProductIndexBuilder.getBatchScopeValues` method determines this range. The map's `START_KEY` specifies the beginning of the range, and the `END_KEY` specifies the conclusion of the range.

 **Note:** This key is available *only* in B2B Commerce for Visualforce Spring '20 (version 4.12) or later.




## Logic Performed

By default, this method applies the following formula for determining the product IDs in `ccrz.ccProductIndexBuilder.BATCH_JOB_PRODUCT_SCOPE`:

```
200 / (Number of locales + 1)
```

What do these values mean?

- The 200 value is an arbitrary default starting point.
- The *Number of locales* variable considers both [specific and generic fallback locales](#). For example, if your storefront defines two specific locales (`en_US` and `ja_JP`), then this variable also considers their respective fallback locales (`en` and `ja`). In this example, the *Number of locales* variable evaluates to 4.
- The + 1 value accounts for a query of `ccrz__E_ProductIndex__c` records.

 **Tip:** In the following B2B Commerce for Visualforce versions, you can override this formula with a specific number of product records to evaluate. This value allows you to avoid Apex governor limits and more precisely control the maximum number of corresponding price list items processed during each product index batch job.

- B2B Commerce for Visualforce **Spring '20 (version 4.12)** patch 3.147.7 or later
- B2B Commerce for Visualforce **Summer '18 (version 4.9)** patch 3.136.39 or later

For more information, see [Control the Size of Each Product Index Batch Job](#).

## Outputs

`Map<String, Object>`, which contains the same input data that was passed to `ccrz.ccProductIndexBuildJob2`.

## Extension Points

You can extend `ccrz.ccProductIndexBuildJob2`, which is a global class that wraps calls to the following methods of the B2B Commerce for Visualforce global API:

### `ccrz.ccApiProductIndex.create`

Creates product index records.

### `ccrz.ccApiProduct.revise`

For each product record related to a product index record, sets the Product Index Status to Pending.

### **Example: Example**

Build product index records for only the `en_US` locale and for only products where Product Index Status is Not Current.

```
ccrz.ccProductIndexBuilder.build(new Map<String, Object> {
    ccrz.ccApiProductIndex.LOCALES => new Set<String>{'en_US', 'en'},
    ccrz.ccApiProductIndex.CLEAN_ONLY => TRUE
});
```

SEE ALSO:

[Salesforce Help Values for Product Index Status](#)

[Salesforce Help Strategies for Managing Price List Item Currencies with Product Index Records](#)

[Salesforce Help Implement the Primary Index for All Currencies Strategy for Product Index Records](#)

## ccrz.ccProductIndexBuilder.disableOld

Executes the `ccrz.ccProductIndexDisableOldJob` batch job, which updates the Active field on existing product index records. All product index records where Active is `FALSE` update so that Active is `TRUE`, and vice versa.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
global static Map<String, Object> disableOld (Map<String, Object>)
```

### Inputs (Optional)

This method accepts the `Map<String, Object>` from `ccrz.ccProductIndexBuildJob2`. The map can specify the following keys:

#### `ccrz.ccApiProductIndex.CLEAN_ONLY`

Boolean

Value	Usage
<code>TRUE</code>	Update product index records only for products where Product Index Status is Pending.
<code>FALSE</code> (default)	Update product index records for all the org's products.

#### `ccrz.ccProductIndexBuilder.BATCH_JOB_PRODUCT_SCOPE`

`Map<String, Object>` that specifies a range of product IDs whose product index records you want to update. The `ccrz.ccProductIndexBuilder.getBatchScopeValues` method determines this range. The map's `START_KEY` specifies the beginning of the range, and the `END_KEY` specifies the conclusion of the range.

 **Note:** This key is available *only* in B2B Commerce for Visualforce Spring '20 (version 4.12).

### Outputs

`Map<String, Object>`, which contains the same input data that was passed to `ccrz.ccProductIndexDisableOldJob`.

## Extension Points

You can extend `ccrz.ccProductIndexDisableOldJob`, which is a global class that wraps calls to the following methods of the B2B Commerce for Visualforce global API:

### `ccrz.ccApiProduct.revise`

For each product record related to a product index record, sets the Product Index Status to Current.

### `ccrz.ccApiProductIndex.revise`

Sets the Active field on each product index record to `TRUE` or `FALSE`, whichever is opposite the current value.

SEE ALSO:

[Salesforce Help Values for Product Index Status](#)

## `ccrz.ccProductIndexBuilder.cleanup`

Executes the `ccrz.ccProductIndexCleanupJob` batch job, which removes existing product index records where the Active field is `FALSE`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> cleanup (Map<String, Object>)
```

## Inputs (Optional)

This method accepts the `Map<String, Object>` from `ccrz.ccProductIndexDisableOldJob`. The map can specify the following keys:

### `ccrz.ccProductIndexBuilder.BATCH_JOB_PRODUCT_SCOPE`

`Map<String, Object>` that specifies a range of product IDs whose product index records you want to remove, if applicable. The `ccrz.ccProductIndexBuilder.getBatchScopeValues` method determines this range. The map's `START_KEY` specifies the beginning of the range, and the `END_KEY` specifies the conclusion of the range.

 **Note:** This key is available *only* in B2B Commerce for Visualforce Spring '20 (version 4.12).

## Outputs

Map<[String](#), [Object](#)>, which contains the same input data that was passed to `ccrz.ccProductIndexCleanupJob`.

## Extension Points

You can extend `ccrz.ccProductIndexCleanupJob`, which is a global class that wraps calls to the following methods of the B2B Commerce for Visualforce global API:

### `ccrz.ccApiProductIndex.remove`

Deletes product index records where the Active field is `FALSE`.

SEE ALSO:

[Salesforce Help Values for Product Index Status](#)

## ccrz.ccSiteIndexBuilder

Builds site index records by executing a series of synchronously executed Apex batch jobs. Each batch job includes a `finish` method that calls back into the `ccrz.ccSiteIndexBuilder` class for starting the next batch job.

## Compatibility

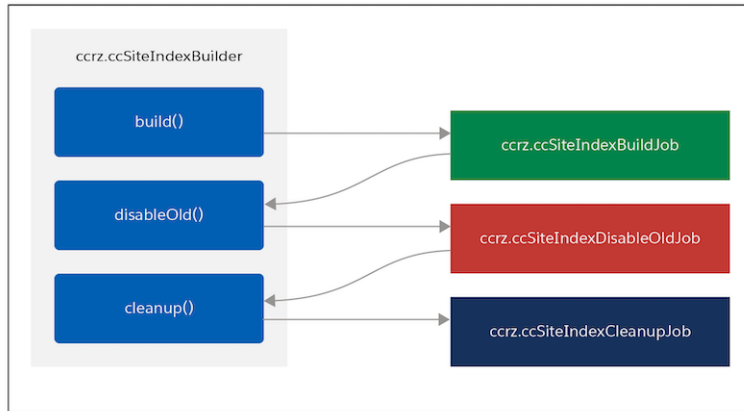
This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Execution Order

The methods of the `ccrz.ccSiteIndexBuilder` class execute when you [refresh site index records from CC Admin](#).

The methods of this class and the related batch job classes execute in the following order.



## Methods

### IN THIS SECTION:

#### [build](#)

Executes the `ccrz.ccSiteIndexBuildJob` batch job for building site index records.

#### [disableOld](#)

Executes the `ccrz.ccSiteIndexDisableOldJob` batch job, which updates the Active field on existing site index records. All site index records where Active is `FALSE` update so that Active is `TRUE`, and vice versa.

#### [cleanup](#)

Executes the `ccrz.ccSiteIndexCleanupJob` batch job, which removes existing site index records where Active is `FALSE`.

## `ccrz.ccSiteIndexBuilder.build`

Executes the `ccrz.ccSiteIndexBuildJob` batch job for building site index records.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> build(Map<String, Object>)
```

## Inputs (Required)

`Map<String, Object>` that must include the following *required* keys:

### `ccrz.ccApiSiteIndex.STOREFRONT`

String that specifies the name of the storefront to create site index records for. The method first queries all the [storefront's supported locales](#), including the default locale.

 **Note:** If the storefront supports the `en_US` locale, the batch job creates site index records for both the full `en_US` locale and the language-only `en` locale.

## Outputs

This method passes the `Map<String, Object>`, updated with the following keys, to `ccrz.ccSiteIndexBuildJob`:

### `ccrz.ccApiSiteIndex.LOCALES`

`Set<String>` of locales to create site index records for.

### `ccrz.ccApiSiteIndex.ROOT_CATEGORY_ID`

String that specifies the ID of the [storefront's root category](#).

### `ccrz.ccApiSiteIndex.STOREFRONT`

String that specifies the name of the storefront to create site index records for.

## Extension Points

You can extend `ccrz.ccSiteIndexBuildJob`, which is a global class that wraps calls to the [ccrz.ccApiSiteIndex.create](#) method.

SEE ALSO:

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

## `ccrz.ccSiteIndexBuilder.disableOld`

Executes the `ccrz.ccSiteIndexDisableOldJob` batch job, which updates the Active field on existing site index records. All site index records where Active is `FALSE` update so that Active is `TRUE`, and vice versa.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> disableOld(Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccSiteIndexBuildJob.finish` method. The map specifies the following keys:

### **ccrz.ccApiSiteIndex.LOCALES**

`Set<String>` of locales whose corresponding site index records the batch job is updating.

### **ccrz.ccApiSiteIndex.STOREFRONT**

String that specifies the name of the storefront whose site index records the batch job is updating.

## Outputs

This method passes the `Map<String, Object>` to `ccrz.ccSiteIndexDisableOldJob`.

## Extension Points

You can extend `ccrz.ccSiteIndexDisableOldJob`, which is a global class that wraps calls to the [ccrz.ccApiSiteIndex.revise](#) method.

SEE ALSO:

[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

## ccrz.ccSiteIndexBuilder.cleanup

Executes the `ccrz.ccSiteIndexCleanupJob` batch job, which removes existing site index records where `Active` is `FALSE`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
global static Map<String, Object> cleanup(Map<String, Object>)
```

## Inputs (Required)

This method accepts the `Map<String, Object>` from the `ccrz.ccSiteIndexDisableOldJob.finish` method. The map specifies the following keys:

### **ccrz.ccApiSiteIndex.LOCALES**

Set<String> of locales whose corresponding site index records the batch job is removing.

### **ccrz.ccApiSiteIndex.STOREFRONT**

String that specifies the name of the storefront whose site index records the batch job is removing.

## Outputs

This method returns the original `Map<String, Object>` input.

## Extension Points

You can extend `ccrz.ccSiteIndexCleanupJob`, which is a global class that wraps calls to the `ccrz.ccApiSiteIndex.remove` method.

SEE ALSO:


[Salesforce Help Select Which Locales and Currencies Your Storefront Supports](#)

## ccrz.ccUtil

Defines a set of static utility methods for evaluating, parsing, or hashing B2B Commerce for Visualforce object data.

This content is available only as an archived PDF from CloudCraze documentation:



(PDF) [ccrz.ccUtil](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## Batch Processing Classes for B2B Commerce for Visualforce

B2B Commerce for Visualforce provides several Apex classes that provide global, asynchronous batch jobs. Each batch processing class extends the global `ccrz.ccBatchJob` base class.



-  **Note:** Sometimes, B2B Commerce for Visualforce updates batch processing classes in a new version of the B2B Commerce managed package. In subscriber code, you reference the latest version of these classes that's available in your version of the managed package.
-  **Tip:** Some batch jobs invoke default logic or data service provider classes, such as `ccrz.ccLogicPICreate` and `ccrz.ccLogicOrderSplitOrder`. You can extend these default service provider classes to customize their behavior, but the batch jobs that invoke them execute only in the context of your org's default storefront. Batch jobs can't execute in the context of a specific storefront. When you extend a default service provider class invoked by a batch job, update the settings for your *default storefront* to override the class that you extended. For information about overriding a default service provider class, see [Extend or Override a Default Logic Service Provider](#) on page 652.

#### IN THIS SECTION:

##### [ccrz.cc\\_batch\\_SubscriptionProcessor](#)

Processes subscriptions and creates orders for subscription installments.

##### [ccrz.ccBatchJob](#)

Defines a global base class that other batch processing classes extend.

##### [ccrz.ccCacheBuilder](#)

Defines a global base class that other classes extend for building caches of specific types of data.

##### [ccrz.ccCategoryCacheBuildJob](#)

Builds a cache of category tree data for specific storefronts, storefront root categories, and locales. This class is called from the `ccrz.ccCategoryCacheBuilder.build` method, which extends the base `ccrz.ccCacheBuilder.build` method.

##### [ccrz.ccPageLabelCacheBuildJob](#)

Builds a cache of page label data for specific storefronts, pages, and locales. This class is called from the `ccrz.ccPageLabelCacheBuilder.build` method, which extends the base `ccrz.ccCacheBuilder.build` method.

##### [ccrz.ccPublicCacheCleanupJob](#)

Deletes disabled cache entries that haven't been modified in the previous seven days. The `ccrz.ccCategoryCacheBuilder.cleanup` and `ccrz.ccPageLabelCacheBuilder.cleanup` methods, which extend the base `ccrz.ccCacheBuilder.cleanup` method, execute this batch job.

##### [ccrz.ccPublicCacheDisableJob](#)

Disables cache entries for specific storefronts, pages, and locales. The `ccrz.ccCategoryCacheBuilder.disableOld` and `ccrz.ccPageLabelCacheBuilder.disableOld` methods, which extend the base `ccrz.ccCacheBuilder.disableOld` method, execute this batch job.

##### [ccrz.ccSplitOrderJob](#)

Provides batch processing for splitting orders.

##### [ccrz.ccSubscriptionRenewalJob](#)

Renews an expired subscription, which is a subscription where the installment or order count has reached zero.

## ccrz.cc\_batch\_SubscriptionProcessor

Processes subscriptions and creates orders for subscription installments.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.cc\\_batch\\_SubscriptionProcessor](#)

**!** **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## ccrz.ccBatchJob

Defines a global base class that other batch processing classes extend.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Methods

None

## ccrz.ccCacheBuilder

Defines a global base class that other classes extend for building caches of specific types of data.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

This class defines the following base methods that don't perform any functionality. Each method accepts a `Map<String, Object>` input and returns a `Map<String, Object>` output.

```
global virtual Map<String, Object> build(Map<String, Object>)
```

```
global virtual Map<String, Object> disableOld(Map<String, Object>)
```

```
global virtual Map<String, Object> cleanup(Map<String, Object>)
```

## ccrz.ccCategoryCacheBuildJob

Builds a cache of category tree data for specific storefronts, storefront root categories, and locales. This class is called from the `ccrz.ccCategoryCacheBuilder.build` method, which extends the base `ccrz.ccCacheBuilder.build` method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Logic Performed

For each combination of storefronts, storefront root categories, and category locales in a batch, this class completes the following actions:

1. Calls the `ccrz.ccApiCategory.fetchCategoryTree` global API method.
2. Deserializes the `ccrz.ccApiCategory.CATEGORYTREE` output.
3. Calls the `ccrz.ccApiPublicCache.startWrite` global API method to initiate the cache build, with the following inputs:

<code>ccrz.ccApiPublicCache.startWrite</code> Key	Value
<code>ccrz.ccApiPublicCache.BUILD_GROUP</code>	The Salesforce ID of the batch job.
<code>ccrz.ccApiPublicCache.CACHE_NAME</code>	<code>ccrz.ccCategoryCacheBuilder.CACHE_NAME</code> , which is always <code>ccCategoryTree</code> .
<code>ccrz.ccApiPublicCache.PAGENAME</code>	<i>all</i>

4. Calls the `ccrz.ccApiPublicCache.appendTo` global API method to add the deserialized category data to the cache entry.

5. Calls the `ccrz.ccApiPublicCache.close` global API method.

When the batch job completes, old cache entries are disabled and, after seven days, deleted.

## Constructors

This class defines the following constructors:

```
ccrz.ccCategoryCacheBuildJob(final Set<Object> storefronts, final Set<Object>
rootCategoryIDs, final Set<Object> locales, final Map<String, Object> inputData)
```

Constructor that takes:

- A set of storefronts to iterate over in the batch.
- A set of storefront root category IDs.
- A set of locales.
- General input data.

SEE ALSO:

[ccrz.ccApiCategory.fetchCategoryTree](#)

[ccrz.ccApiPublicCache](#)

[Salesforce Help Organize Products into Categories for B2B Commerce for Visualforce](#)

[Salesforce Help Improve Storefront Performance with Platform Cache](#)

## ccrz.ccPageLabelCacheBuildJob

Builds a cache of page label data for specific storefronts, pages, and locales. This class is called from the `ccrz.ccPageLabelCacheBuilder.build` method, which extends the base `ccrz.ccCacheBuilder.build` method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Logic Performed

For each combination of storefronts, pages, and locales in a batch, this class completes the following actions:

1. Calls the `ccrz.ccApiI18N.fetchPageLabels` global API method.
2. Parses the `ccrz.ccApiI18N.fetchPageLabels` return data for the `ccrz.ccApiI18N.PAGE_LABELS` output.

3. Calls the `ccrz.ccApiPublicCache.startWrite` global API method to initiate the cache build, with the following inputs:

<code>ccrz.ccApiPublicCache.startWrite</code> Key	Value
<code>ccrz.ccApiPublicCache.BUILD_GROUP</code>	The Salesforce ID of the batch job.
<code>ccrz.ccApiPublicCache.CACHE_NAME</code>	<code>ccrz.ccPageLabelCacheBuilder.CACHE_NAME</code> , which is always <code>ccPageLabel</code> .
<code>ccrz.ccApiPublicCache.PAGENAME</code>	The storefront page for the current batch scope.

4. Calls the `ccrz.ccApiPublicCache.appendTo` global API method to add the `ccrz.ccApiI18N.PAGE_LABELS` data to the cache entry.
5. Calls the `ccrz.ccApiPublicCache.close` global API method.

When the batch job completes, the `ccrz.ccPageLabelCacheBuilder.disableOld` method executes. This method ensures that only a single page label cache entry is enabled at one time for each combination of storefront, page, and locale.

## Constructors

This class defines the following constructors:

**`ccrz.ccPageLabelCacheBuildJob`**(**final** **Set**<**Object**> *storefronts*, **final** **Set**<**Object**> *pages*, **final** **Set**<**Object**> *locales*, **final** **Map**<**String**, **Object**> *inputData*)

Constructor that takes:

- A set of storefronts to iterate over in the batch.
- A set of storefront pages.
- A set of locales.
- General input data that's passed to both the `ccrz.ccPageLabelCacheBuilder.disableOld` and `ccrz.ccPageLabelCacheBuilder.cleanup` methods.

## Limits

This job doesn't support a combination of storefront name, page name, or locale values that exceeds 130,000 characters. This limit is the maximum length of the `ccrz__Content__c` field on the `ccrz__E_PublicCacheContent__c` object.

To verify the length of a particular combination of values, execute the following Anonymous Apex code from the Developer Console:

```
Map<String, Object> pageLabelRet = ccrz.ccApiI18N.fetchPageLabels(new Map<String, Object>{
    ccrz.ccApi.API_VERSION => ccrz.ccAPI.CURRENT_VERSION,
    ccrz.ccApiI18N.STORE_NAME => 'storefront',
    ccrz.ccApiI18N.PAGE_NAME => 'pagename',
    ccrz.ccApiI18N.LOCALE => 'locale',
    ccrz.ccApiI18N.USE_CACHE => false,
    ccrz.ccApiI18N.RETURN_SERIAL => true
});
```

```
String pageLabels = (String) pageLabelRet.get('pageLabels');
System.debug(pageLabels.length());
```

SEE ALSO:

[ccrz.ccApil18N.fetchPageLabels](#)

[ccrz.ccApiPublicCache](#)

[Salesforce Help Customize Storefront Headings and Text with Page Labels](#)

[Salesforce Help Improve Storefront Performance with Platform Cache](#)

## ccrz.ccPublicCacheCleanupJob

Deletes disabled cache entries that haven't been modified in the previous seven days. The `ccrz.ccCategoryCacheBuilder.cleanup` and `ccrz.ccPageLabelCacheBuilder.cleanup` methods, which extend the base `ccrz.ccCacheBuilder.cleanup` method, execute this batch job.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Constructors

This class defines the following constructors:

**`ccrz.ccPublicCacheCleanupJob(final String cacheName, final Map<String, Object> inputData)`**

Constructor that takes:

- A cache name, such as `ccCategoryTree` or `ccPageLabel`.
- A map of general input data, which by default isn't evaluated.

**`ccrz.ccPublicCacheCleanupJob(final String cacheName, final Integer maxAge, final Map<String, Object> inputData)`**

Constructor that takes:

- A cache name, such as `ccCategoryTree` or `ccPageLabel`.
- An integer that specifies a maximum age, in days. This batch job deletes only the cache entries that haven't been modified during this time. The default value is 7.
- A map of input data to pass to:
  - [ccrz.ccApiPublicCache.disableOld](#)

- `ccrz.ccCategoryCacheBuilder.cleanup`
- `ccrz.ccPageLabelCacheBuilder.cleanup`

SEE ALSO:

[ccrz.ccApiPublicCache.disableOld](#)

## ccrz.ccPublicCacheDisableJob

Disables cache entries for specific storefronts, pages, and locales. The `ccrz.ccCategoryCacheBuilder.disableOld` and `ccrz.ccPageLabelCacheBuilder.disableOld` methods, which extend the base `ccrz.ccCacheBuilder.disableOld` method, execute this batch job.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Logic Performed

For each combination of storefronts, pages, and locales in a batch, this class calls the `ccrz.ccApiPublicCache.disableOld` global API method.

## Constructors

This class defines the following constructors:

**`ccrz.ccPublicCacheDisableJob(final Set<Object> storefronts, final Set<Object> pages, final Set<Object> locales, final String cacheName, final String builderClassName, final Map<String, Object> inputData)`**

Constructor that takes:

- A set of storefronts to iterate over in the batch.
- A set of storefront pages.
- A set of locales.
- A cache name, such as `ccCategoryTree` or `ccPageLabel`.
- A cache builder class that invoked this batch job, such as `ccCategoryCacheBuilder` or `ccPageLabelCacheBuilder`. When this class name is included, this batch job calls the class' `cleanup` method after this batch job completes.

- A map of input data to pass to:
  - `ccrz.ccApiPublicCache.disableOld`
  - `ccrz.ccCategoryCacheBuilder.cleanup`
  - `ccrz.ccPageLabelCacheBuilder.cleanup`

SEE ALSO:

[ccrz.ccApiPublicCache.disableOld](#)

## ccrz.ccSplitOrderJob

Provides batch processing for splitting orders.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Logic Performed

This batch processing class completes the following actions:

1. Calls the `ccrz.ccApiOrder.getOrdersToSplit` global API method, which returns the `ccrz__E_Order__c` records to split. By default, this method splits any order where the `ccrz__SplitProcessed__c` field is `FALSE`.
2. Calls the `ccrz.ccApiOrder.splitOrder` global API method to execute the split.

### Constructors

This class defines the following constructors:

**`ccrz.ccSplitOrderJob`(final String storefront)**

Constructor that specifies the name of the storefront to split orders for.

SEE ALSO:

[ccrz.ccApiOrder.getOrdersToSplit](#)

[ccrz.ccApiOrder.splitOrder](#)



## ccrz.ccSubscriptionRenewalJob

Renews an expired subscription, which is a subscription where the installment or order count has reached zero.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [ccrz.ccSubscriptionRenewalJob](#)

**Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

## Trust Classes for B2B Commerce for Visualforce

---

B2B Commerce for Visualforce provides some default Apex classes for authoring unit tests and logging execution data.

**Note:** Sometimes, B2B Commerce for Visualforce updates trust classes in a new version of the B2B Commerce managed package. In subscriber code, you reference the latest version of these classes that's available in your version of the managed package.

IN THIS SECTION:

[ccrz.ccApiTestData](#)

Create data for B2B Commerce for Visualforce unit tests. This class defines a single method, `setUpData`, that accepts a `Map<String, Map<String, Object>>` or serialized JSON string of the data to create.

[ccrz.ccLog](#)

Write code execution information to the console for developing, troubleshooting, or debugging your storefront.

### ccrz.ccApiTestData

Create data for B2B Commerce for Visualforce unit tests. This class defines a single method, `setUpData`, that accepts a `Map<String, Map<String, Object>>` or serialized JSON string of the data to create.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [Global Test Data](#)

**Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

### ccrz.ccLog

Write code execution information to the console for developing, troubleshooting, or debugging your storefront.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## General Usage Guidelines

Consider the following guidelines when choosing how to invoke `ccrz.ccLog` from your subscriber code:

- Write only the minimum log lines required for debugging an issue.
- Avoid log lines within loops or within methods called from loops, especially tight, computational loops. Instead, log the inputs to the loop and the loop's results.
- When you log a method's entry, also log the method's exit, even when the method throws an exception. Review the `ccrz.ccLog` examples, which use pairs of `try` and `finally` code blocks.
- When you log a method's entry or exit, log only the top-level methods, such as the initial remote action call or top-level business logic method.

## Logging Level Conventions

Each `ccrz.ccLog` method takes one of the following `System.LoggingLevel` enum values as an input parameter:

Enum Value	Usage
<code>System.LoggingLevel.DEBUG</code>	For execution status, input data to a method, or other troubleshooting information.
<code>System.LoggingLevel.ERROR</code>	For exceptions caught in a <code>catch</code> block.
<code>System.LoggingLevel.INFO</code>	For high-level processes, such as a method's entry or exit.
<code>System.LoggingLevel.WARN</code>	For unexpected data, behavior, or system status. For example, use <code>System.LoggingLevel.WARN</code> when a method returns a null value or other unexpected return data parameters.

 **Note:** The `ccrz.ccLog` methods don't support other enum values that `System.LoggingLevel` defines.

## Subject Conventions

Each `ccrz.ccLog` method can take a string input parameter that specifies a subject for the log line. Use the following conventions for the subject's string, depending on the specific information that you're logging:

Data to Log	Subject Value
Exception	<code>ERR</code>

Data to Log	Subject Value
Method's entry	<i>M:E</i>
Method's exit	<i>M:X</i>
Method's parameter	<i>P:parameterName</i>
Method's return value	<i>R</i>
Other data	<i>D:dataToLog</i> , where <i>dataToLog</i> is a string that's specific to the type of data you're logging. Or, simply use <i>D</i> .
Custom subscriber code	To distinguish custom code from default code included in the managed package, consider including a custom prefix such as <i>S</i> . For example, to log a custom method's entry, use <i>SM:X</i> .
Local code for development only, and not intended for production	A common subject or prefix that's easy to search for, such as <i>CUSTOM_M:X</i> .

 **Note:**

- Because the colon character (:) is a delimiter in the subject line, avoid using a colon as part of a custom prefix.
- The `ccrz.cLog` methods support up to three levels (two colon delimiters) in a subject, such as *A:B:C*. However, we recommend that you use only two levels (one colon delimiter) whenever possible, such as *A:B*.
- Avoid concatenating strings for the subject line.


## Properties

This class defines the following Apex `global static` property. For more information about Apex properties and accessors, see [Apex Properties](#).

**logOn {get; set;}**

Boolean

Value	Usage
<i>true</i>	Logging is enabled for the current browser session.
<i>false</i>	Logging isn't enabled.

 **Note:** If your logging call requires complex calculations, [check the value of this property before you execute other logic](#).

## Message Methods

This class defines the following methods for writing string messages to the log output. Consider the following conventions when writing strings to the log output:

- Avoid concatenating strings.

- Use minimal language for the message. For example, avoid the following verbose message:

```
ccrz.ccLog.log(System.LoggingLevel.WARN, 'D:someParameter', 'Something unexpected occurred! The expected data doesn't exist.')
```

Instead, use the following minimal message:

```
ccrz.ccLog.log(System.LoggingLevel.WARN, 'D:someParameter', 'someParameter is null')
```

- Avoid method calls or other logic within the message, such as:

```
ccrz.ccLog.log(System.LoggingLevel.ERROR, 'D:someParameter', someMethodThatGetsMessage())
```

- To construct a complex message, wrap the construction within a check of the `logOn` property. If the message construction can throw an exception, include the appropriate `try` and `catch` code blocks. See the [logOn property example](#).

**global static void log(System.LoggingLevel level, String subject, String message)**

Writes a log line with:

- `System.LoggingLevel` enum value that you specify
- `Subject` that you specify
- Message that you specify

**global static void log(System.LoggingLevel level, String message)**

Writes a log line with:

- `System.LoggingLevel` enum value that you specify
- Message that you specify

**global static void log(String subject, String message)**

Writes a log line with:

- `System.LoggingLevel.DEBUG` enum value by default
- `Subject` that you specify
- Message that you specify

**global static void log(String message)**

Writes a log line with:

- `System.LoggingLevel.DEBUG` enum value by default
- Message that you specify

## Object Methods


This class defines the following methods for writing object data to the log output. Consider the following conventions when logging object data:

- Be careful when logging large amounts of data, which affect [Apex governor limits for heap size](#).
- Be careful about the size of the object data that you're logging. Log only the values used within the current process, rather than the complete object.
- Avoid logging any personal identifiable information (PII) or other sensitive data.
- Avoid logging any data on the server that the client already logs, such as `ccrz_cc.RemoteActionContext` and its results.

**global static void log(System.LoggingLevel level, String subject, Object object)**

Writes a log line with:

- `System.LoggingLevel` enum value that you specify
- `Subject` that you specify
- Object data, serialized to a string using `JSON.serializePretty`

 **Note:** If JSON serialization fails, this method falls back to a `String.valueOf` representation of the object data. If logging is disabled, this method passes the object and `System.LoggingLevel` enum value directly to `System.debug`.

**global static void log(System.LoggingLevel level, Object object)**

Writes a log line with:

- `System.LoggingLevel` enum value that you specify
- Object data that you specify

**global static void log(String subject, Object object)**

Writes a log line with:

- `System.LoggingLevel.DEBUG` enum value by default
- `Subject` that you specify
- Object data that you specify

**global static void log(Object object)**

Writes a log line with:

- `System.LoggingLevel.DEBUG` enum value by default
- Object data that you specify

## Exception Methods

This class defines the following methods for writing exceptions to the log output:

**global static void log(System.LoggingLevel level, String subject, Exception e)**

Writes a log line with:

- `System.LoggingLevel` enum value that you specify
- `Subject` that you specify
- Exception caught, which invokes the following methods:
  1. `e.getMessage`
  2. `e.getStackTraceString`

**global static void log(System.LoggingLevel level, Exception e)**

Writes a log line with:

- `System.LoggingLevel` enum value that you specify
- Exception caught

**global static void log(String subject, Exception e)**

Writes a log line with:

- `System.LoggingLevel.DEBUG` enum value by default
- `Subject` that you specify
- Exception caught


**global static void log(Exception e)**

Writes a log line with:

- `System.LoggingLevel.DEBUG` enum value by default
- Exception caught

### Example: Example: Check the Value of the `logOn` Property

Check the value of the `logOn` property, and execute other logic only when `logOn` is `true`.

 **Important:** When logging is enabled, you can call other methods that query or process object data. However, we recommend that you avoid SOQL or DML operations, which can cause your org to reach Apex governor limits.

```
if(ccrz.ccLog.logOn) {
    try {
        String someString = '';
        ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'D:someString', someString);
    } catch (Exception loggingError) {
        ccrz.ccLog.log(System.LoggingLevel.ERROR, 'ERR', loggingError);
    }
}
```

### Example: Example: Log from Remote Action Methods

When you call `ccrz.ccLog` methods within an Apex remote action, make sure that you implement the following requirements:

- Call `ccrz.cc_CallContext.initRemoteContext` before you call `ccrz.ccLog`.
- Call `ccrz.cc_RemoteActionResult` at the end of the remote action.

```
@RemoteAction
global static ccrz.cc_RemoteActionResult exampleMethod (ccrz.cc_RemoteActionContext
ctx, String exampleParameter) {
    ccrz.cc_CallContext.initRemoteContext(ctx);
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:E', 'exampleMethod');
    ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:exampleParameter', exampleParameter);

    ccrz.cc_RemoteActionResult result = new ccrz.cc_RemoteActionResult();
    result.success = false;
    try {
        // Execute business logic
        result.success = true;
    } catch (Exception e) {
        ccrz.ccLog.log(System.LoggingLevel.ERROR, 'ERR', e);
    } finally {
        ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X', 'exampleMethod');
        ccrz.ccLog.close(result);
    }
    return result;
}
```

### Example: Example: Log from a Boolean Return Apex Method

The following example shows how to call `ccrz.ccLog` methods within the `try` block of an Apex method that returns a Boolean result.

```
global static Boolean exampleMethod (final String parameter1, final Integer parameter2,
final Object parameter3) {
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:E', 'exampleMethod');
```


```

ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter1', parameter1);
ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter2', parameter2);
ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter3', parameter3);
Boolean result = false;
try {
    try {
        Integer someInteger = 1;
        Integer anotherInteger = 2;
        // Execute business logic
        ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'D:someInteger', someInteger);
        // Continue business logic
        ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'D:anotherInteger',
anotherInteger);
    } catch (Exception e) {
        ccrz.ccLog.log(System.LoggingLevel.ERROR, 'ERR', e);
    }
    return result;
} finally {
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'R', result);
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X', 'exampleMethod');
}
}

```

### Example: Example: Log from a Boolean Return Apex Method, but Don't Catch Exceptions

The following example shows how to call `ccrz.ccLog` methods within the `try` block of an Apex method that returns a Boolean result. This example doesn't catch any exceptions that occur within the `try` block, and only returns the result.

 **Note:** In this example, the method doesn't log the return value, but always logs the method's exit. With this design, ensure that the calling method logs and handles exceptions.

```

global static Boolean exampleMethod (final String parameter1, final Integer parameter2,
final Object parameter3) {
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:E', 'exampleMethod');
    ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter1', parameter1);
    ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter2', parameter2);
    ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter3', parameter3);
    Boolean result = false;
    try {
        Integer someInteger = 1;
        Integer anotherInteger = 2;
        // Execute business logic
        ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'D:someInteger', someInteger);
        // Continue business logic
        ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'D:anotherInteger', anotherInteger);

        return result;
    } finally {
        ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X', 'exampleMethod');
    }
}

```


### Example: Example: Log from a Void Return Apex Method

The following example shows how to call `ccrz.ccLog` methods within the `try` block of an Apex method that doesn't return any value.

```
global static void exampleMethod (final String parameter1, final Decimal parameter2)
{
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:E', 'exampleMethod');
    ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter1', parameter1);
    ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter2', parameter2);
    try {
        String someString = null;
        // Execute business logic
        ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'D:someString', someString);
    } catch (Exception e) {
        ccrz.ccLog.log(System.LoggingLevel.ERROR, 'ERR', e);
    } finally {
        ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X', 'exampleMethod');
    }
}
```

### Example: Example: Log from a Void Return Apex Method, but Don't Catch Exceptions

The following example shows how to call `ccrz.ccLog` methods within the `try` block of an Apex method that doesn't return any value. This example doesn't catch any exceptions that occur within the `try` block.

 **Note:** With this design, ensure that the calling method logs and handles exceptions.

```
global static void exampleMethod (final String parameter1, final Decimal parameter2)
{
    ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:E', 'exampleMethod');
    ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter1', parameter1);
    ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'P:parameter2', parameter2);
    try {
        String someString = null;
        // Execute business logic
        ccrz.ccLog.log(System.LoggingLevel.DEBUG, 'D:someString', someString);
    } finally {
        ccrz.ccLog.log(System.LoggingLevel.INFO, 'M:X', 'exampleMethod');
    }
}
```

#### SEE ALSO:

[Salesforce Help Log Storefront Code Execution](#)

`debug(logLevel, msg)`

`serializePretty(objectToSerialize)`

[Exception Class and Built-In Exceptions](#)



# JavaScript Functions for B2B Commerce for Visualforce

---

B2B Commerce for Visualforce provides some proprietary JavaScript libraries or heavily modified third-party libraries that help render components and behavior on storefront pages. As part of your storefront's customization strategy, you can invoke, extend, or override these functions.

The `ccrz.cc_hk_UserInterface` extension point class in the API includes methods that return `<script>` element paths to the following libraries:

## CCRZ Includes

The `ccrz.cc_hk_UserInterface.ccrzIncludes` method returns paths to the following proprietary or heavily modified third-party libraries:

- `jquery.currency.js`
- `cloudcraze.js`
- `CC_Backbone.js`
- `backbone_models.js`
- `analytics.js`

## Theme-specific Includes

The `ccrz.cc_hk_UserInterface.standardUIProperties` method returns paths to the following proprietary libraries:

- The default `uiproperties.js` file, installed with your storefront's `CC_JavaScript_Framework` static resource.
- A theme-specific override of `uiproperties.js` included in your [storefront custom theme's static resource](#).

## Namespace

All functions in these libraries are part of the global CCRZ JavaScript namespace.

When you create custom functions that extend or override the default functions, append the `subsc` sub-namespace, as in `CCRZ.subsc`. We recommend that any subscriber function definitions first check to validate that the `subsc` object exists, and create the object if it's missing:

```
CCRZ.subsc = _.extend(CCRZ.subsc||{}, {
  // Subscriber-defined attributes and functions here
  customFunction : function() {
    alert('customFunction says hello');
  }
});
```



**Warning:** Avoid creating any other child objects in the CCRZ namespace in your subscriber code.

## Other Third-Party Libraries

B2B Commerce for Visualforce also references other unmodified or lightly modified third-party JavaScript libraries. The `ccrz.cc_hk_UserInterface.standardIncludes` method returns paths to these third-party libraries. These libraries are outside of the global CCRZ namespace. For a complete list, see [Default JavaScript Libraries for Storefront Pages](#).

## IN THIS SECTION:

[CC\\_Backbone.js](#)

The `CC_Backbone.js` file, installed with your storefront's `CC_JavaScript_Framework` static resource, contains objects that help render components on a page.

[cloudcraze.js](#)

The `cloudcraze.js` file, installed with your storefront's `CC_JavaScript_Framework` static resource, contains objects that help render components on a page.

[uiproperties.js](#)

The `uiproperties.js` file, installed with your storefront's `CC_JavaScript_Framework` static resource, contains a single object, `CCRZ.uiProperties`.

[JavaScript Functions and Objects for Localized Storefront Content](#)

When you internationalize your storefront, B2B Commerce for Visualforce uses JavaScript functions and objects defined in `cloudcraze.js` for showing localized content.

[Handlebars Functions and Objects for Storefront Page Templates](#)

B2B Commerce for Visualforce uses JavaScript functions and objects defined in `cloudcraze.js` for calling `handlebars.js` functionality that renders page content.

## SEE ALSO:

[ccrz.cc\\_hk\\_UserInterface.ccrzIncludes](#)

[ccrz.cc\\_hk\\_UserInterface.standardIncludes](#)

[ccrz.cc\\_hk\\_UserInterface.standardUIProperties](#)

## CC\_Backbone.js

The `CC_Backbone.js` file, installed with your storefront's `CC_JavaScript_Framework` static resource, contains objects that help render components on a page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## IN THIS SECTION:

[CCRZ.RemoteInvocation](#)

Represents a base object that wraps Apex remote action methods for persisting `CCRZ.pagevars.remoteContext` information.

[CCRZ.CloudCrazeModel](#)

Extends the `CCRZ.RemoteInvocation` base object and the default `Backbone.Model` object. Individual components, such as the Breadcrumb widget and Category Tree widget, extend this object.

[CCRZ.CloudCrazeView](#)

Extends the `CCRZ.RemoteInvocation` base object and the default `Backbone.View` object. Individual components, such as the line-level independence (LLI) checkout experience, extend this object for rendering content and responding to changes in device screen size.

[PubSub](#)

Represents an event channel that extends `Backbone.Events` and provides a central, synchronous channel across all views and models.

## SEE ALSO:

[ccrz.cc\\_hk\\_UserInterface.ccrzIncludes](#)

[Salesforce Help Code That Renders Storefront Components](#)

[External Link Backbone.js](#)

## CCRZ.RemoteInvocation

Represents a base object that wraps Apex remote action methods for persisting `CCRZ.pagevars.remoteContext` information.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Example Apex Remote Action Method

Let's say that you've defined the following subscriber class, `exampleSubscriberClass`, with a remote action method, `exampleRemoteActionMethod`:

```
global class exampleSubscriberClass {
    @RemoteAction
    global static ccrz.cc_RemoteActionResult
exampleRemoteActionMethod(ccrz.cc_RemoteActionContext ctx, String exampleInputString) {
    ccrz.cc_RemoteActionResult result = ccrz.cc_CallContext.init(ctx);
    try {
        // Execute business logic for exampleInputString
        result.success = true;
    } catch (Exception e) {
```

```

        ccrz.ccLog.log(LoggingLevel.ERROR, 'Err', e);
        // Exception handling
    } finally {
        ccrz.ccLog.close(result);
    }
    return result;
}
}

```

## Wrap the Apex Remote Action Method

Functions that wrap an Apex remote action method—including the default functions and any subscriber functions you author that extend `CCRZ.RemoteInvocation`—comply with the following design considerations:

- The wrapped remote action method must accept a `ccrz.cc_RemoteActionContext` as its first parameter before any other optional parameters.
- The wrapping functions include a callback that's invoked when the remote action returns. The callback passes the following parameters:

### **result**

JSON serialization of `ccrz.cc_RemoteActionResult`.

### **event**

Event object that describes the status of the remote action. See [Handling the Remote Response](#).

- The wrapping functions can include `options`, a JavaScript object that contains [configuration parameters for the remote action](#). For example, the `buffer` parameter can indicate whether the remote action can be combined with other remote action requests. The `CCRZ.RemoteInvocation` object extends `options` to include the parameter `namespace`, which defines the `ccrz` namespace for the remote action call. By default, this configuration restricts remote actions in a function to those that the managed package defines. In subscriber code, you can extend the `namespace` parameter to reference a different managed package or set its value to `false`.
- Functions that extend `CCRZ.RemoteInvocation` must specify the `className` parameter, which references the underlying Apex class that defines the wrapped remote action method. In the previous example, this class is `exampleSubscriberClass`.

## Methods

### IN THIS SECTION:

#### [invokeCtx](#)

Invokes a remote action method for the current `ccrz.cc_RemoteActionContext` and other parameters.

[invokePageLoadingCtx](#)

Invokes a remote action method for the current `ccrz.cc_RemoteActionContext` and other parameters as a page is loading. This function causes an overlay to appear, or increment the overlay count when an overlay already appears. When the function completes, the overlay count decrements, and the overlay disappears when the count is zero.

## SEE ALSO:

[ccrz.cc\\_RemoteActionContext](#)[ccrz.cc\\_RemoteActionResult](#)[Declaring a Remote Method in Apex](#)[Salesforce Help Code That Renders Storefront Components](#)[External Link Backbone.js](#)

## CCRZ.RemoteInvocation.invokeCtx

Invokes a remote action method for the current `ccrz.cc_RemoteActionContext` and other parameters.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8



### Example: Example

Subscriber code defines an object, `customRemoteInvocation`, that:

- Extends `CCRZ.RemoteInvocation`
- Defines a method, `exampleFunction`, that:
  - Calls `invokeCtx()` for wrapping the remote action method, `exampleRemoteActionMethod`.
  - Sets the `buffer` parameter to `false`, which indicates that the remote action executes by itself.
  - Sets the `nmsp` parameter to `false`, which indicates that the remote action method is subscriber code from outside of the `ccrz` namespace.

```
CCRZ.subsc.customRemoteInvocation = __.extend(CCRZ.RemoteInvocation, {
  className: 'remoteActionApexClass',
  exampleFunction: function(exampleInputString) {
    this.invokeCtx('exampleRemoteActionMethod', exampleInputString, function(result,
    event) {
      if(event.status) {
```

```

        if(result && result.success) {
            // Execute logic when the result is successful
        } else {
            // Handle an unsuccessful result
        }
    } else {
        // Handle a default remote action failure
    }
},
{
    buffer: false
    nmsp: false
});
}
});

```

### CCRZ.RemoteInvocation.invokePageLoadingCtx

Invokes a remote action method for the current `ccrz.cc_RemoteActionContext` and other parameters as a page is loading. This function causes an overlay to appear, or increment the overlay count when an overlay already appears. When the function completes, the overlay count decrements, and the overlay disappears when the count is zero.

#### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### CCRZ.CloudCrazeModel

Extends the `CCRZ.RemoteInvocation` base object and the default `Backbone.Model` object. Individual components, such as the Breadcrumb widget and Category Tree widget, extend this object.

#### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

This object doesn't define any methods.

SEE ALSO:

[External Link Backbone.Model](#)

## CCRZ.CloudCrazeView

Extends the `CCRZ.RemoteInvocation` base object and the default `Backbone.View` object. Individual components, such as the line-level independence (LLI) checkout experience, extend this object for rendering content and responding to changes in device screen size.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Parameters

This object defines the following parameters:

### **viewName**

Boolean

Value	Usage
<code>true</code>	Publishes the <code>CCRZ.pubSub</code> event channel, <code>view: <b>viewName</b>: refresh</code> , after the view is rendered.
<code>false</code> (default)	Doesn't publish any event channel.

**managedSubView**

Boolean

Value	Usage
<i>true</i>	The view doesn't automatically re-render when the browser window size or device screen size changes. When this value is <i>true</i> , the view expects another view or object to specify re-render behavior.
<i>false</i> (default)	The view automatically re-renders when the browser window size or device screen size changes.

## Create a View

When the `CCRZ.util.createView` method instantiates a `CCRZ.CloudCrazeView` object, the `CCRZ.CloudCrazeView` methods execute in the following order:

1. `initialize`
2. `render`
3. `preViewChanged`
4. `preRender`
5. `preRenderPhone`, `preRenderTablet`, or `preRenderDesktop`
6. `renderPhone`, `renderTablet`, or `renderDesktop`
7. `postRenderPhone`, `postRenderTablet`, or `postRenderDesktop`
8. `postRender`

## Handle a Changed View Size

When the `viewChanged` method returns `true`, the `CCRZ.CloudCrazeView` methods execute in the following order:

1. `renderViewChanged`
2. `render`
3. `preViewChanged`
4. `preRender`
5. `preRenderPhone`, `preRenderTablet`, or `preRenderDesktop`
6. `renderPhone`, `renderTablet`, or `renderDesktop`
7. `postRenderPhone`, `postRenderTablet`, or `postRenderDesktop`
8. `postRender`

## Methods

### IN THIS SECTION:

#### [initialize](#)

Creates a view before the view renders on the page.



### [render](#)

Renders the view for any browser window or device screen size. This method calls into the render methods for specific device screen sizes and handles fallback behavior.

### [preViewChanged](#)

Provides a placeholder for executing logic before the browser window size or device screen size changes.

### [preRender](#)

Provides a placeholder for executing logic before the view renders on the page. This method executes for any browser window or device screen size.

### [preRenderPhone](#)

Provides a placeholder for executing logic before the `renderPhone` method renders the view for mobile device screen sizes.

### [preRenderTablet](#)

Provides a placeholder for executing logic before the `renderTablet` method renders the view for tablet device screen sizes.

### [preRenderDesktop](#)

Provides a placeholder for executing logic before the `renderDesktop` method renders the view for desktop device screen sizes.

### [renderPhone](#)

Renders the view specifically for mobile device screen sizes. If the view doesn't define this method but the current viewport is a mobile device screen size, the view falls back to the `renderTablet` and then `renderDesktop` methods.

### [renderTablet](#)

Renders the view specifically for tablet device screen sizes. If the view doesn't define this method but the current viewport is a tablet device screen size, the view falls back to the `renderDesktop` method.

### [renderDesktop](#)

Renders the view specifically for desktop device screen sizes. The view must define this method, which provides fallback rendering for all device screen sizes.

### [postRenderPhone](#)

Provides a placeholder for executing logic after the `renderPhone` method renders the view for mobile device screen sizes.

### [postRenderTablet](#)

Provides a placeholder for executing logic after the `renderTablet` method renders the view for tablet device screen sizes.

### [postRenderDesktop](#)

Provides a placeholder for executing logic after the `renderDesktop` method renders the view for desktop device screen sizes.

### [postRender](#)

Provides a placeholder for executing logic after the view renders on the page. This method executes for any browser window or device screen size.

### [viewChanged](#)

Listens for changes in browser window or device screen size or orientation, and calls `CCRZ.CloudCrazeView.renderViewChanged` when a change occurs.

### [renderViewChanged](#)

Calls `CCRZ.CloudCrazeView.render` to re-render the view when `CCRZ.CloudCrazeView.viewChanged` returns `true`.

#### SEE ALSO:

[External Link Backbone.View](#)

## CCRZ.CloudCrazeView.initialize

Creates a view before the view renders on the page.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
initialize: function(options)
```

### Inputs

#### **options**

JavaScript object that contains [configuration parameters for the remote action](#). The `CCRZ.RemoteInvocation` object extends `options` to include the parameter `nmsp`, which defines the `CCRZ` namespace for the remote action call.

## CCRZ.CloudCrazeView.render

Renders the view for any browser window or device screen size. This method calls into the render methods for specific device screen sizes and handles fallback behavior.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
render: function()
```

## Inputs

None

## Outputs

CCRZ.CloudCrazeView object for the rendered view.

## CCRZ.CloudCrazeView.preViewChanged

Provides a placeholder for executing logic before the browser window size or device screen size changes.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
preViewChanged: function(oldView, newView)
```

## Inputs

### **oldView**

Backbone.View object that represents the view before the browser window size or device screen size changes.

### **newView**

Backbone.View object that represents the view after the browser window size or device screen size changes.

SEE ALSO:

[External Link Backbone.View](#)

## CCRZ.CloudCrazeView.preRender

Provides a placeholder for executing logic before the view renders on the page. This method executes for any browser window or device screen size.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
preRender: function ()
```

## Inputs

None

## CCRZ.CloudCrazeView.preRenderPhone

Provides a placeholder for executing logic before the `renderPhone` method renders the view for mobile device screen sizes.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
preRenderPhone: function ()
```

## Inputs

None

## CCRZ.CloudCrazeView.preRenderTablet

Provides a placeholder for executing logic before the `renderTablet` method renders the view for tablet device screen sizes.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
preRenderTablet: function()
```

### Inputs

None

## CCRZ.CloudCrazeView.preRenderDesktop

Provides a placeholder for executing logic before the `renderDesktop` method renders the view for desktop device screen sizes.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
preRenderDesktop: function()
```

## Inputs

None

## CCRZ.CloudCrazeView.renderPhone

Renders the view specifically for mobile device screen sizes. If the view doesn't define this method but the current viewport is a mobile device screen size, the view falls back to the `renderTablet` and then `renderDesktop` methods.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
renderPhone: function()
```

## Inputs

None

## CCRZ.CloudCrazeView.renderTablet

Renders the view specifically for tablet device screen sizes. If the view doesn't define this method but the current viewport is a tablet device screen size, the view falls back to the `renderDesktop` method.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
renderTablet: function()
```

## Inputs

None

## CCRZ.CloudCrazeView.renderDesktop

Renders the view specifically for desktop device screen sizes. The view must define this method, which provides fallback rendering for all device screen sizes.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
renderDesktop: function()
```

## Inputs

None

## CCRZ.CloudCrazeView.postRenderPhone

Provides a placeholder for executing logic after the `renderPhone` method renders the view for mobile device screen sizes.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
postRenderPhone: function()
```

## Inputs

None

## CCRZ.CloudCrazeView.postRenderTablet

Provides a placeholder for executing logic after the `renderTablet` method renders the view for tablet device screen sizes.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
postRenderTablet: function()
```

## Inputs

None

## CCRZ.CloudCrazeView.postRenderDesktop

Provides a placeholder for executing logic after the `renderDesktop` method renders the view for desktop device screen sizes.

## Compatibility

This reference applies to:



Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
postRenderDesktop: function()
```

## Inputs

None

## CCRZ.CloudCrazeView.postRender

Provides a placeholder for executing logic after the view renders on the page. This method executes for any browser window or device screen size.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
postRender: function()
```

## Inputs

None

## CCRZ.CloudCrazeView.viewChanged

Listens for changes in browser window or device screen size or orientation, and calls `CCRZ.CloudCrazeView.renderViewChanged` when a change occurs.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

### Signature

```
viewChanged: function()
```

### Inputs

None

### Outputs

Boolean

Value	Usage
true	The current browser window or device screen size or orientation changed. Call <a href="#">CCRZ.CloudCrazeView.renderViewChanged</a> .
false	The current browser window or device screen size or orientation hasn't changed, and a re-render isn't necessary.

## CCRZ.CloudCrazeView.renderViewChanged

Calls `CCRZ.CloudCrazeView.render` to re-render the view when `CCRZ.CloudCrazeView.viewChanged` returns `true`.

### Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
renderViewChanged: function ()
```

## Inputs


None

## CCRZ.PubSub

Represents an event channel that extends `Backbone.Events` and provides a central, synchronous channel across all views and models.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [CloudCraze Event Handling](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[External Link Backbone.Events](#)

## cloudcraze.js

The `cloudcraze.js` file, installed with your storefront's `CC_JavaScript_Framework` static resource, contains objects that help render components on a page.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**IN THIS SECTION:**[CCRZ.console](#)

Provides debugging output to the browser console when the Developer Mode storefront setting is enabled.

[CCRZ.display](#)

Defines methods that return information about the device currently used for showing the storefront.

[CCRZ.util](#)

Defines utility methods that other B2B Commerce for Visualforce JavaScript methods reference.

## CCRZ.console

Provides debugging output to the browser console when the Developer Mode storefront setting is enabled.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

**IN THIS SECTION:**[log](#)

Writes output to the browser console.

## CCRZ.console.log

Writes output to the browser console.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
CCRZ.console.log(message)
```

## Inputs

*message*

The output to write to the developer console.

## Outputs

None

## CCRZ.display

Defines methods that return information about the device currently used for showing the storefront.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

IN THIS SECTION:

[isDesktop](#)

Indicates whether the current view state is a desktop device size.

**isPhone**

Indicates whether the current view state is a phone device size.

**isTablet**

Indicates whether the current view state is a tablet device size.

**CCRZ.display.isDesktop**

Indicates whether the current view state is a desktop device size.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

**Signature**

```
CCRZ.display.isDesktop()
```

**Inputs**

None

**Outputs**

Boolean

Value	Usage
true	The current view state is a desktop device size.
false	The current view state is a different device size.

**CCRZ.display.isPhone**

Indicates whether the current view state is a phone device size.

**Compatibility**

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
CCRZ.display.isPhone ()
```

## Inputs

None

## Outputs

Boolean

Value	Usage
true	The current view state is a phone device size.
false	The current view state is a different device size.

## CCRZ.display.isTablet

Indicates whether the current view state is a tablet device size.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
CCRZ.display.isTablet()
```

## Inputs

None

## Outputs

Boolean

Value	Usage
true	The current view state is a tablet device size.
false	The current view state is a different device size.

## CCRZ.util

Defines utility methods that other B2B Commerce for Visualforce JavaScript methods reference.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Methods

IN THIS SECTION:

[createView](#)

Constructs and instantiates a `CCRZ.CloudCrazeView` object. This method minimizes the amount of code required to create a view, especially when a `Backbone.Model` or `Backbone.Collection` isn't required.

### CCRZ.util.createView

Constructs and instantiates a `CCRZ.CloudCrazeView` object. This method minimizes the amount of code required to create a view, especially when a `Backbone.Model` or `Backbone.Collection` isn't required.



## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## Signature

```
CCRZ.util.createView(viewConfiguration)
```

## Inputs

### **viewConfiguration**

JavaScript object that represents the view to create for different device screen sizes:

#### **desktop**

Specifies the following details for desktop device screen sizes:

##### **template**

The `handlebars.js` template ID for desktop device screen sizes.

##### **target**

The HTML class for rendering content on desktop device screen sizes.

#### **model**

`CCRZ.CloudCrazeModel` object passed to the `handlebars.js` template for rendering content.

#### **phone**

Specifies the following details for mobile device screen sizes:

##### **template**

The `handlebars.js` template ID for mobile device screen sizes.

##### **target**

The HTML class for rendering content on mobile device screen sizes.

#### **tablet**

Specifies the following details for tablet device screen sizes:

##### **template**

The `handlebars.js` template ID for tablet device screen sizes.

##### **target**

The HTML class for rendering content on tablet device screen sizes.

### Note:

- This method evaluates specific views in the following fallback order:

1. Phone
  2. Tablet
  3. Desktop
- This object can also define view-specific functions or invoke `CCRZ.CloudCrazeView` parameters and methods.

### Example: Example

Consider the following example component that renders a `handlebars.js` template named `exampleTemplate`.

```
<div class="exampleTarget"/>
<script id="exampleTemplate" type="text/template">
<p>This is an example template</p>
</script>

<script>
  jQuery(function($) {
    CCRZ.util.createView({
      desktop: {
        template: 'exampleTemplate',
        target: 'exampleTarget'
      },
      phone: {
        template: 'exampleTemplate',
        target: 'phone-center-column'
      },
      preRender: function() {
        alert('Text to show before the content renders');
      }
    });
  });
</script>
```

SEE ALSO:

[CCRZ.CloudCrazeView](#)

## uiproperties.js

The `uiproperties.js` file, installed with your storefront's `CC_JavaScript_Framework` static resource, contains a single object, `CCRZ.uiProperties`.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

#### IN THIS SECTION:

##### [CCRZ.uiProperties](#)

Defines `handlebars.js` templates and partials, HTML selectors, and some default properties for components on storefront pages.

#### SEE ALSO:

[ccrz.cc\\_hk\\_UserInterface.standardUIProperties](#)

[Salesforce Help Code That Renders Storefront Components](#)

[Salesforce Help Style Your Storefront with a Theme](#)

## CCRZ.uiProperties

Defines `handlebars.js` templates and partials, HTML selectors, and some default properties for components on storefront pages.

## Compatibility

This reference applies to:

Release	Managed Package Version	API Version
B2B Commerce for Visualforce Winter '21	4.13	12
B2B Commerce for Visualforce Spring '20	4.12	11
B2B Commerce for Visualforce Summer '19	4.11	10
B2B Commerce for Visualforce Spring '19	4.10	9
B2B Commerce for Visualforce Summer '18	4.9	8

## `handlebars.js` Component Templates and Selectors

The `CCRZ.uiProperties` object defines the following default information for each storefront component, such as the header, page widgets, page details, and more:

### Desktop

- `handlebars.js` template to use for rendering the component on desktop devices
- Unique HTML class or ID selector in the page source where the component renders on desktop devices


### Phone

- `handlebars.js` template to use for rendering the component on mobile devices

- Unique HTML class or ID selector in the page source where the component renders on mobile devices

For example, consider an example component:

```
exampleComponentView: {
  desktop: {
    tmpl: 'example-Component-View-Desktop',
    selector: '.htmlClass'
  },
  phone: {
    tmpl: 'example-Component-View-Mobile',
    selector: '#htmlID'
  }
}
```

 **Note:** The `phone` template and selector information is an artifact of legacy Bootstrap 2.3.2-enabled storefront design, which specified different templates for desktop and mobile devices. We recommend that new solutions always use the Bootstrap 3 library, where a component's template is responsive for any device screen size. If you [extend the default `uiproperties.js` file](#), define only `desktop` template and selector information, and ensure that your template is responsive for any device screen size.

## Remove Mobile-Specific Views

If your storefront uses the Bootstrap 3-enabled templates (recommended), you can optionally remove the mobile-specific template and selector information from the `CCRZ.uiProperties` object. The following example script modifies all `phone` template and selector information to use the same `desktop` information:

```
_.each(CCRZ.uiProperties, function(data) {
  if (data.desktop && data.phone) {
    data.phone = data.desktop;
  } else if (data.desktop && data.tablet) {
    data.tablet = data.desktop;
  } else {
    //console.log(data);
    _.each(data, function(data) {
      if (data.desktop && data.phone) {
        data.phone = data.desktop;
      } else if (data.desktop && data.tablet) {
        data.tablet = data.desktop;
      }
    });
  }
});
```

## handlebars.js Component Partial

Some components use `handlebars.js` [partials](#), which are registered within `CCRZ.uiProperties` and define reusable template sections. A partial's template name must use the `#` character prefix. Consider a similar example component, which defines partials for the component's features, such as controls that a buyer interacts with or other sections:

```
exampleComponentView: {
  desktop: {
    tmpl: 'example-Component-View-Desktop',
    selector: '.htmlClass'
  },
}
```

```

phone: {
  tmpl: 'example-Component-View-Mobile',
  selector: '#htmlID'
},
partials: {
  componentControl: '#ComponentDisplay',
  componentSection: '#ComponentSection',
  componentSummary: '#ComponentSummary'
}
}

```

## Autocomplete Properties

The `CCRZ.uiProperties` object defines default properties for autocomplete suggestions for the following components:

- [Search Box widget](#) in the storefront header or in a page column in the page body
- [Quick Order widget](#)

```

autocomplete: {
  PRODUCT_SEARCH: {
    search_input: {
      minLength: 3,
      delay: 300,
      resultLimit: 10,
      position: {collision: 'none flip'}
    },
    sidebarSearchText: {
      minLength: 3,
      delay: 300,
      resultLimit: 10,
      position: {collision: 'none flip'}
    }
  },
  QUICK_ORDER: {
    quickOrderField: {
      minLength: 3,
      delay: 300,
      resultLimit: 10,
      position: {collision: 'none flip'}
    }
  }
}
}

```

These properties define the following behavior:

### **collision**

Option in `jquery-ui.js` that describes how overflowing elements render on the page. For more information, see the `jquery-ui.js` [documentation for the position option](#). This value is passed to the jQuery code that renders the suggestions on the page.

### **delay**

Time (ms) before suggestions appear on the page after a buyer has entered the required minimum characters in the widget. This value is passed to the jQuery code that renders the suggestions on the page.

**minLength**

The minimum number of characters that a buyer must enter in the widget before suggestions are returned. This value is passed to the jQuery code that renders the suggestions on the page.

**resultLimit**

The maximum number of suggested results to show on the page. This value is passed to the server as part of the autocomplete request.

**Download the Default `uiproperties.js` File**

To review all the information in the default `uiproperties.js`, download the `CC_JavaScript_Framework` static resource.

1. From Setup, enter *Static Resources* in the Quick Find box, and then select **Static Resources**.
2. Select **CC\_JavaScript\_Framework**.
3. In the static resource detail, click **View file** to download it to your local system.
4. Open the downloaded ZIP folder and copy the contents to a safe working directory on your local system.

**Extend the Default `CCRZ.uiProperties` Values**

As part of your storefront's customization strategy, you can override values in the `CCRZ.uiProperties` object in the following ways:

**Override from your theme**

Add specific override values in the `uiproperties.js` file included in your [storefront theme's static resource](#).

**Note:**

- Overrides from your theme's `uiproperties.js` file apply globally to all storefront pages.
- In your override, include only the specific values that you want to change.

For example, the following theme-specific `uiproperties.js` file overrides the default autocomplete `minLength` value:

```
CCRZ.uiProperties = $.extend(true, CCRZ.uiProperties, {
  autoComplete: {
    QUICK_ORDER: {
      quickOrderField: {
        minLength: 5
      }
    }
  }
})
```

**Override from a page include**

Add override values in a [page include](#).

**Important:**

- Make sure that you specify override values before any jQuery `onload` functions execute.
- Make sure that the default `uiproperties.js` file's definitions load before your override's definitions load.

For example, the following page include also overrides the default autocomplete `minLength` value:

```
// Other page include code
CCRZ.uiProperties.autoComplete.QUICK_ORDER.quickOrderField.minLength = 5;
```

```
// Other page include code, such as:  
jQuery(function($) {  
    CCRZ.pubSub.on()  
});
```

## Considerations for `CCRZ.uiProperties` Upgrades

Major releases of the B2B Commerce for Visualforce managed package can sometimes add definitions to the default `CCRZ.uiProperties` object. Generally, these updates don't affect any existing subscriber extensions. Occasionally, some updates can require that you manually upgrade the default `CCRZ.uiProperties` object. For more information about any specific required action, review the [B2B Commerce for Visualforce ReleaseNotes and Upgrade Guide](#).



**Tip:** We recommend that you always upgrade the managed package in a sandbox org and review the updated `uiproperties.js` file before you upgrade your solution in a production org.

### SEE ALSO:

[ccrz.cc\\_hk\\_UserInterface.standardUIProperties](#)

[Salesforce Help Code That Renders Storefront Components](#)

[Salesforce Help Style Your Storefront with a Theme](#)

[Salesforce Help Considerations for Autocomplete Suggestions in Search Box and Quick Order Widgets](#)

[External Link Handlebars](#)

## JavaScript Functions and Objects for Localized Storefront Content

When you internationalize your storefront, B2B Commerce for Visualforce uses JavaScript functions and objects defined in `cloudcraze.js` for showing localized content.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [Localization Functions](#)



**Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

### SEE ALSO:


[Salesforce Help Internationalize Your Storefront](#)

## Handlebars Functions and Objects for Storefront Page Templates

B2B Commerce for Visualforce uses JavaScript functions and objects defined in `cloudcraze.js` for calling `handlebars.js` functionality that renders page content.

This content is available only as an archived PDF from CloudCraze documentation:

(PDF) [Handlebars Functions](#)

 **Important:** The content of this PDF was authored for older versions of the managed package and doesn't necessarily represent the most recent version of B2B Commerce for Visualforce. This content is updated infrequently, and only for resolving critical inaccuracies or changes in behavior.

SEE ALSO:

[Salesforce Help Code That Renders Storefront Components](#)