
Insurance Developer Guide

Version 64.0, Summer '25

Summer '25



CONTENTS

Chapter 1: Get Started with Insurance Developer Resources	1
Chapter 2: Insurance Product Administration	2
Insurance Product Administration Business APIs	3
Resources	3
Requests	10
Responses	20
Chapter 3: Insurance Quoting	23
Insurance Quoting Business APIs	24
Resources	24
Requests	38
Responses	57
Insurance Quoting Standard Invocable Actions	63
Create Insurance Quote Action	63
Get Insurance Quote Details Action	66
Reprice Insurance Product Action	67
Insurance Quoting Developer Journey	69
Build a Self-service Quote and Policy Creation Functionality for Community Portal Users	69
Calculate Quote Price Without Creating a Formal Quote	70
Implement Quote Creation Functionality for Insurance Representatives	70
Implement Quote Update Functionality for Insurance Representatives	70
Chapter 4: Insurance Policy Administration	71
Insurance Policy Business APIs	72
Resources	72
Requests	80
Responses	86
Insurance Policy Standard Invocable Actions	92
Cancel Insurance Policy Action	93
Endorse Insurance Policy Action	94
Get Insurance Policy Action	97
Issue Insurance Policy Action	98
Renew Insurance Policy Action	100
Renew Insurance Policies Action	102
Insurance Policy Administration Metadata API Types	104
Flow for Insurance Policy Administration	104
Insurance Policy Administration Apex Reference	105
ConnectApi Namespace	105

Contents

Insurance Policy Developer Journey	117
Create and Manage Insurance Policies by Using Salesforce Quoting	117
Create and Manage Insurance Policies Without Using Salesforce Quoting	120
Cancel Policy	122
Chapter 5: Insurance Brokerage	124
Insurance Brokerage Business APIs	125
Resources	126
Requests	136
Responses	145
Insurance Brokerage Standard Invocable Actions	168
Compute Producer Splits Action	168
Create Producer Commissions Action	170
Find Insurance Policy Action	172
Insurance Brokerage Metadata API Types	175
Flow for Insurance Brokerage	175
Insurance Brokerage Apex Reference	175
ConnectApi Namespace	176

CHAPTER 1 Get Started with Insurance Developer Resources

Use the developer resources of Insurance Cloud to manage end-to-end insurance processes. Create and configure quotes, manage policy transactions, and automate processes related to brokerage policies and commissions.

Learn more about the developer resources that are available for these components.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise, Unlimited, and Developer** Editions

Insurance Product Administration

Define the products you sell to customers, including their classifications, categories, and attributes. Create customized price adjustment methods and pricing procedures. Define rules that determine how your products behave when they're quoted, sold, and serviced. See [Insurance Product Administration](#) on page 2.

Insurance Quoting

Automate quoting experiences as part of a cohesive customer-centric experience. Create, update, configure, and delete quotes and quote line items. See [Insurance Quoting](#) on page 23.

Insurance Policy Administration

Manage policy lifecycles across quoting, issuance, endorsements, billing, and renewals. See [Insurance Policy Administration](#) on page 71.

Insurance Brokerage

Manage insurance policy lifecycles. Automate processes related to Policy and Commissions Management. See [Insurance Brokerage](#) on page 124.

CHAPTER 2 Insurance Product Administration

In this chapter ...

- [Insurance Product Administration Business APIs](#)

Use the Insurance Product Administration developer resources to define the products you sell to customers, including their classifications, categories, and attributes. Create customized price adjustment methods and pricing procedures. Define rules that determine how your products behave when they're quoted, sold, and serviced.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise, Unlimited, and Developer** Editions

Insurance Product Administration Business APIs

Design, manage, and administer insurance products to meet the business requirements.

This table lists the available Insurance Product Administration resources.

Resource	Description
/connect/insurance/underwriting-rules (POST)	Create an underwriting rule for a root product based on the selected state definition, transition, and object name.
/connect/insurance/underwriting-rule-group-status/uwrGroupId (PATCH)	Update the status of an underwriting rule group.
/connect/insurance/underwriting-rules/underwritingRuleId (PATCH)	Update an existing underwriting rule for a root product.

Resources

Learn more about the available Insurance Product Administration API resources.

Requests

Learn more about the available Insurance Product Administration requests.

Responses

Learn more about the available response bodies of Insurance Product Administration APIs.

Resources

Learn more about the available Insurance Product Administration API resources.

Insurance Underwriting Rules (POST)

Create an underwriting rule for a root product based on the selected state definition, transition, and object name.

Insurance Underwriting Rule Group Status (PATCH)

Update the status of an underwriting rule group.

Insurance Underwriting Rules (Patch)

Update an existing underwriting rule for a root product.

Insurance Underwriting Rules (POST)

Create an underwriting rule for a root product based on the selected state definition, transition, and object name.

Resource

```
/connect/insurance/underwriting-rules
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/underwriting-rules
```

Available version

63.0

HTTP methods

POST

Request body for POST**JSON example**

```
{
  "name": "Automatically Approve Quote",
  "apiName": "approveQuoteBasedonComprehensiveCoverageDeductibles",
  "effectiveFromDate": "2024-05-30T10:30:00",
  "effectiveToDate": "2024-11-01T10:30:00",
  "status": "Draft",
  "description": "Rule description",
  "underwritingRuleGroup": {
    "underwritingRuleGroupId": "1KQ000000MKJIK", // Optional,
    // if underwritingRuleGroupId exist, then use it as parent instead of create
    a new record.
    "stageTransitionId": "0jjDR00000000VGAAA", // State Transition id.
    "stageTransitionName": "Draft to Approved",
    "objectType": "Quote",
    "recordType": "--Master--",
    "fromStage": "Draft",
    "toStage": "Approved",
    "rootProductId": "01tDR000000EMslyAG" // Auto Gold.
    "criteriaExpression": "1" // Exmple "(1 AND 2) OR 3" if more than 1 rules.
  }
  "productPath": "01tDR000000EMslyAG/11BDR00000000T72AI", // "Auto Gold/Auto
Vehicle"
  "sequence": 1,
  "ruleCriteria": [
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Annual Milage",
          "attributePicklistValueId": null,
          "type": "Attribute",
          "attributeId": "0tjDR00000000VGAY",
          "dataType": "Number",
          "values": [
            "50000"
          ]
        }
      ],
      "rootObjectId": "01tDR000000EMslyAG", // Auto Gold.
      "productRelatedComponent" : // Need More details on it.
      "criteriaIndex": 1,
      "sourceContextTagName": "ProductBasedOn",
      "sourceOperator": "Equals",
      "sourceDataType": "String",

```



```

    "sourceValues": [
      "11BDR00000000T72AI" // Auto Vehicle.
    ],
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Comprehensive Deductible",
          "type": "Attribute",
          "attributePicklistValueId": null,
          "attributeId": "0tjDR00000000UaYAI",
          "dataType": "Number",
          "values": [
            "$5000"
          ]
        }
      ],
      "rootObjectId": "01tDR0000000EMslYAG", // Auto Bundle
      "productRelatedComponent" : // Need More details on it.
      "criteriaIndex": 2,
      "sourceContextTagName": "Product",
      "sourceOperator": "Equals",
      "sourceDataType": "String",
      "sourceValues": [
        "01tDR0000000EMsoYAG" // Comprehensive Product.
      ]
    },
    // EvaluationSuccessTaskGroup and EvaluationFailureTaskGroup are optional
    // Fulfillment plan design time entities will be created.
    "evaluationSuccessTaskGroup": // Create FulfillmentStepDefinitionGroup records

    {
      "name": "QuoteApproveSuccessTaskGroup",
      "tasks": [ // create FulfillmentStepDefinition records
        {
          "name": "KYC check flow", // FulfillmentStepDefinition API Names
          "type": "AutoTask", // FulfillmentStepDefinition type
          "flowApiName": "KYC check flow", // Flow API Names
        }
      ],
    },
    "evaluationFailureTaskGroup": //Similar to EvaluationSuccessTaskGroup
    {
      "name": "QuoteApproveFailureTaskGroup",
      "tasks": [
        {
          "name": "Vehicle ineligible",
          "type": "AutoTask",
          "flowApiName": "Mail Notification - Vehicle ineligible",
        }
      ]
    }

```

```

    ]
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
apiName	String	Unique API name of the underwriting rule.	Required	63.0
description	String	Description of the underwriting rule	Optional	63.0
effectiveFromDate	String	Date from which the underwriting rule is effective.	Required	63.0
effectiveToDate	String	Date until which the underwriting rule is effective.	Optional	63.0
evaluationFailedTaskGroup	Evaluation Task Group Input[]	Input details for failed evaluation tasks.	Optional	63.0
evaluationSuccessfulTaskGroup	Evaluation Task Group Input[]	Input details for successful evaluation tasks.	Optional	63.0
name	String	Name of the underwriting rule	Required	63.0
productPath	String	Name of the product path.	Required	63.0
ruleCriteria	Insurance Rule Criteria Input[]	Rule criteria for the underwriting rule.	Required	63.0
status	String	Status of the underwriting rule. Possible values are: <ul style="list-style-type: none"> • Active • Inactive • Draft 	Optional	63.0
underwritingRuleGroup	Insurance Underwriting Rule Group Input[]	The UnderwritingRuleGroup input data.	Required	63.0

Response body for POST[Insurance Create Underwriting Rules](#)**Insurance Underwriting Rule Group Status (PATCH)**

Update the status of an underwriting rule group.

Resource

```
/connect/insurance/underwriting-rule-group-status/uwrGroupId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/underwriting-rule-group-status/umwGroupId
```

Available version

63.0

HTTP methods

PATCH

Request body for PATCH**JSON example**

```
{
  "status": "Active"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
status	String	The status of the underwriting rule group.	Required	63.0

Response body for PATCH[Insurance Underwriting Rule Group Update Status](#)

Insurance Underwriting Rules (Patch)

Update an existing underwriting rule for a root product.

Resource

```
/connect/insurance/underwriting-rules/underwritingRuleId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/underwriting-rules/underwritingRuleId
```

Available version

63.0

HTTP methods

PATCH

Request body for PATCH**JSON example**

```
{
  "name": "Automatically Approve Quote",
  "apiName": "approveQuoteBasedonComprehensiveCoverageDeductibles",
  "effectiveFromDate": "2024-05-30T10:30:00",
  "effectiveToDate": "2024-11-01T10:30:00",
  "productPath": "01tDR0000000EMslyAG/11BDR000000000T72AI", // "Auto Gold/Auto Vehicle"
```

```

    "ruleCriteria": [ // ruleCriteria will be replaced as a whole. Updating only
one condition is not supported
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Annual Milage",
          "attributePicklistValueId": null,
          "type": "Attribute",
          "attributeId": "0tjDR00000000VGYAY",
          "dataType": "Number",
          "values": [
            "50000"
          ]
        }
      ],
      "rootObjectId": "01tDR000000EMslYAG", // Auto Gold.
      "productRelatedComponent" : // Need More details on it.
      "criteriaIndex": 1,
      "sourceContextTagName": "ProductBasedOn",
      "sourceOperator": "Equals",
      "sourceDataType": "String",
      "sourceValues": [
        "11BDR00000000T72AI" // Auto Vehicle.
      ]
    },
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Comprehensive Deductible",
          "type": "Attribute",
          "attributePicklistValueId": null,
          "attributeId": "0tjDR00000000UaYAI",
          "dataType": "Number",
          "values": [
            "$5000"
          ]
        }
      ],
      "rootObjectId": "01tDR000000EMslYAG", // Auto Bundle
      "productRelatedComponent" : // Need More details on it.
      "criteriaIndex": 2,
      "sourceContextTagName": "Product",
      "sourceOperator": "Equals",
      "sourceDataType": "String",
      "sourceValues": [
        "01tDR000000EMsoYAG" // Comprehensive Product.
      ]
    }
  ]
}

```

```

],
// EvaluationSuccessTaskGroup and EvaluationFailureTaskGroup are optional
// Fulfillment plan design time entities will be created.
"EvaluationSuccessTaskGroup": // Create FulfillmentStepDefinitionGroup records

{
  "name": "QuoteApproveSuccessTaskGroup",
  "tasks": [ // create FulfillmentStepDefinition records
    {
      "name": "KYC check flow", // FulfillmentStepDefinition API Names
      "type": "AutoTask", // FulfillmentStepDefinition type
      "flowApiName": "KYC check flow", // Flow API Names
    }
  ]
},
"EvaluationFailureTaskGroup": //Similar to EvaluationSuccessTaskGroup
{
  "name": "QuoteApproveFailureTaskGroup",
  "tasks": [
    {
      "name": "Vehicle ineligible",
      "type": "AutoTask",
      "flowApiName": "Mail Notification - Vehicle ineligible",
    }
  ]
}
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
description	String	Description of the underwriting rule.	Optional	63.0
effectiveFromDate	String	Date from which the underwriting rule is effective.	Optional	63.0
effectiveToDate	String	Date until which the underwriting rule is effective.	Optional	63.0
evaluationFailureTaskGroup	Evaluation Task Group Input[]	Input details for failed evaluation tasks.	Optional	63.0
evaluationSuccessTaskGroup	Evaluation Task Group Input[]	Input details for successful evaluation tasks.	Optional	63.0
name	String	Name of the underwriting rule.	Optional	63.0
productPath	String	Name of the product path.	Optional	63.0
ruleCriteria	Insurance Rule Criteria Input[]	Rule criteria for the underwriting rule.	Required	63.0

Name	Type	Description	Required or Optional	Available Version
status	String	Status of the underwriting rule. Possible values are: <ul style="list-style-type: none"> Active Inactive Draft 	Optional	63.0

Response body for PATCH[Insurance Update Underwriting Rules](#)

Requests

Learn more about the available Insurance Product Administration requests.

[Evaluation Task Group Input](#)

Input representation details of the evaluation task group.

[Evaluation Task Input](#)

Input representation details of the evaluation task.

[Insurance Create Underwriting Rules Input](#)

Create Underwriting Rule Input Representation

[Insurance Rule Condition Input](#)

Input representation details of insurance rule condition.

[Insurance Rule Criteria Input](#)

Input representation details of the insurance rule criteria.

[Insurance Underwriting Rule Group Input](#)

Input representation details of the insurance underwriting rule group.

[Insurance Underwriting Rule Group Update Status Input](#)

Input representation of the request to update insurance underwriting rule group status.

[Insurance Update Underwriting Rules Input](#)

Update Underwriting Rule Input Representation

Evaluation Task Group Input

Input representation details of the evaluation task group.

JSON example

```
// EvaluationSuccessTaskGroup and EvaluationFailureTaskGroup are optional
// Fulfillment plan design time entities will be created.
"evaluationSuccessTaskGroup": // Create FulfillmentStepDefinitionGroup records
{
  "name": "QuoteApproveSuccessTaskGroup",
```

```

"tasks": [ // create FulfillmentStepDefinition records
{
"name": "KYC check flow", // FulfillmentStepDefinition API Names
"type": "AutoTask", // FulfillmentStepDefinition type
"flowApiName": "KYC check flow", // Flow API Names
}
],
"evaluationFailureTaskGroup": //Similar to EvaluationSuccessTaskGroup
{
"name": "QuoteApproveFailureTaskGroup",
"tasks": [
{
"name": "Vehicle ineligible",
"type": "AutoTask",
"flowApiName": "Mail Notification - Vehicle ineligible",
}
]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
name	String	Name of the evaluation task group.	Required	63.0
tasks	Evaluation Task Input[]	List of tasks to be executed for this evaluation task group.	Required	63.0

Evaluation Task Input

Input representation details of the evaluation task.

JSON example

```

"tasks": [
{
"name": "Vehicle ineligible",
"type": "AutoTask",
"flowApiName": "Mail Notification - Vehicle ineligible",
}
]

```

Properties

Name	Type	Description	Required or Optional	Available Version
flowApiName	String	API name of the associated flow when the task type is AutoTask..	Required	63.0
name	String	Name of the evaluation task.	Required	63.0

Name	Type	Description	Required or Optional	Available Version
type	String	Type of the evaluation task. Possible values are: <ul style="list-style-type: none"> • AutoTask • Manual 	Required	63.0

Insurance Create Underwriting Rules Input

Create Underwriting Rule Input Representation

JSON example

```
{
  "name": "Automatically Approve Quote",
  "apiName": "approveQuoteBasedonComprehensiveCoverageDeductibles",
  "effectiveFromDate": "2024-05-30T10:30:00",
  "effectiveToDate": "2024-11-01T10:30:00",
  "status": "Draft",
  "description": "Rule description",
  "underwritingRuleGroup": {
    "underwritingRuleGroupId": "1KQ000000MKJIK", // Optional,
    // if underwritingRuleGroupId exist, then use it as parent instead of create a
    new record.
    "stageTransitionId": "0jjDR00000000VGAAA", // State Transition id.
    "stageTransitionName": "Draft to Approved",
    "objectType": "Quote",
    "recordType": "--Master--",
    "fromStage": "Draft",
    "toStage": "Approved",
    "rootProductId": "01tDR000000EMslyAG" // Auto Gold.
    "criteriaExpression": "1" // Exmple "(1 AND 2) OR 3" if more than 1 rules.
  }
  "productPath": "01tDR000000EMslyAG/11BDR00000000T72AI", // "Auto Gold/Auto Vehicle"

  "sequence": 1,
  "ruleCriteria": [
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Annual Milage",
          "attributePicklistValueId": null,
          "type": "Attribute",
          "attributeId": "0tjDR00000000VGAY",
          "dataType": "Number",
          "values": [
            "50000"
          ]
        }
      ]
    }
  ]
}
```



```

    }
  ],
  "rootObjectId": "01tDR000000EMslYAG", // Auto Gold.
  "productRelatedComponent" : // Need More details on it.
  "criteriaIndex": 1,
  "sourceContextTagName": "ProductBasedOn",
  "sourceOperator": "Equals",
  "sourceDataType": "String",
  "sourceValues": [
    "11BDR00000000T72AI" // Auto Vehicle.
  ]
},
{
  "conditions": [
    {
      "contextTagName": "SalesTransactionItemAttribute",
      "operator": "LessThan",
      "conditionIndex": 1,
      "attributeName": "Comprehensive Deductible",
      "type": "Attribute",
      "attributePicklistValueId": null,
      "attributeId": "0tjDR00000000UaYAI",
      "dataType": "Number",
      "values": [
        "$5000"
      ]
    }
  ],
  "rootObjectId": "01tDR000000EMslYAG", // Auto Bundle
  "productRelatedComponent" : // Need More details on it.
  "criteriaIndex": 2,
  "sourceContextTagName": "Product",
  "sourceOperator": "Equals",
  "sourceDataType": "String",
  "sourceValues": [
    "01tDR000000EMsoYAG" // Comprehensive Product.
  ]
}
],
// EvaluationSuccessTaskGroup and EvaluationFailureTaskGroup are optional
// Fulfillment plan design time entities will be created.
"evaluationSuccessTaskGroup": // Create FulfillmentStepDefinitionGroup records
{
  "name": "QuoteApproveSuccessTaskGroup",
  "tasks": [ // create FulfillmentStepDefinition records
    {
      "name": "KYC check flow", // FulfillmentStepDefinition API Names
      "type": "AutoTask", // FulfillmentStepDefinition type
      "flowApiName": "KYC check flow", // Flow API Names
    }
  ]
},
"evaluationFailureTaskGroup": //Similar to EvaluationSuccessTaskGroup
{

```

```

    "name": "QuoteApproveFailureTaskGroup",
    "tasks": [
      {
        "name": "Vehicle ineligible",
        "type": "AutoTask",
        "flowApiName": "Mail Notification - Vehicle ineligible",
      }
    ]
  }
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
apiName	String	Unique API name of the underwriting rule.	Required	63.0
description	String	Description of the underwriting rule	Optional	63.0
effectiveFromDate	String	Date from which the underwriting rule is effective.	Required	63.0
effectiveToDate	String	Date until which the underwriting rule is effective.	Optional	63.0
evaluationFailureTaskGroup	Evaluation Task Group Input[]	Input details for failed evaluation tasks.	Optional	63.0
evaluationSuccessTaskGroup	Evaluation Task Group Input[]	Input details for successful evaluation tasks.	Optional	63.0
name	String	Name of the underwriting rule	Required	63.0
productPath	String	Name of the product path.	Required	63.0
ruleCriteria	Insurance Rule Criteria Input[]	Rule criteria for the underwriting rule.	Required	63.0
status	String	Status of the underwriting rule. Possible values are: <ul style="list-style-type: none"> • Active • Inactive • Draft 	Optional	63.0
underwritingRuleGroup	Insurance Underwriting Rule Group Input[]	The UnderwritingRuleGroup input data.	Required	63.0

Insurance Rule Condition Input

Input representation details of insurance rule condition.

JSON example

```

"conditions": [
  {
    "contextTagName": "SalesTransactionItemAttribute",
    "operator": "LessThan",
    "conditionIndex": 1,
    "attributeName": "Annual Milage",
    "attributePicklistValueId": null,
    "type": "Attribute",
    "attributeId": "0tjDR00000000VGyAY",
    "dataType": "Number",
    "values": [
      "50000"
    ]
  }
]

```

Properties

Name	Type	Description	Required or Optional	Available Version
attributeId	String	ID of the attribute.	Required	63.0
attributeName	String	Name of the attribute.	Required	63.0
attributePicklistValueId	String	ID of the attribute picklist value.	Required	63.0
conditionIndex	Integer	Index of the rule condition.	Required	63.0
contextTagName	String	Tag name of the context.	Required	63.0
dataType	String	Data type of the attribute.	Required	63.0
operator	String	Type of operator. Possible values are: <ul style="list-style-type: none"> Equals LessThan 	Required	63.0
type	String	Type of rule condition.	Required	63.0
values	String[]	List of values.	Required	63.0

Insurance Rule Criteria Input

Input representation details of the insurance rule criteria.

JSON example

```

"ruleCriteria": [
  {
    "conditions": [
      {
        "contextTagName": "SalesTransactionItemAttribute",
        "operator": "LessThan",

```

```

    "conditionIndex": 1,
    "attributeName": "Annual Milage",
    "attributePicklistValueId": null,
    "type": "Attribute",
    "attributeId": "0tjDR00000000VGYAY",
    "dataType": "Number",
    "values": [
      "50000"
    ]
  }
]

```

Properties

Name	Type	Description	Required or Optional	Available Version
conditions	Insurance Rule Condition Input[]	List of rule conditions.	Optional	63.0
criteriaIndex	Integer	Index of the rule criteria.	Required	63.0
rootObjectId	String	ID of the root object.	Required	63.0
sourceContextTagName	String	Name of the source context tag.	Required	63.0
sourceDataType	String	Source data type. Possible values are: <ul style="list-style-type: none"> String Integer 	Required	63.0
sourceOperator	String	Source operator. Possible value: <ul style="list-style-type: none"> Equals 	Required	63.0
sourceValues	String[]	List of source values.	Required	63.0

Insurance Underwriting Rule Group Input

Input representation details of the insurance underwriting rule group.

JSON example

```

{
  "underwritingRuleGroup": {
    "underwritingRuleGroupId": "1KQ000000MKJIK", // Optional,
    // if underwritingRuleGroupId exist, then use it as parent instead of create a
    // new record.
    "stageTransitionId": "0jjDR00000000VGAAA", // State Transition id.
    "stageTransitionName": "Draft to Approved",
    "objectType": "Quote",
    "recordType": "--Master--",
    "fromStage": "Draft",
    "toStage": "Approved",
    "rootProductId": "01tDR00000000EMSLYAG" // Auto Gold.
  }
}

```

```
"criteriaExpression": "1" // Exmple "(1 AND 2) OR 3" if more than 1 rules.
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
criteriaExpression	String	Criteria Expression for the underwriting rule group. For example, (1 AND 2) OR 3.	Required	63.0
fromStage	String	Name of the stage the record is moving from.	Optional	63.0
objectType	String	Object type for which the underwriting rule group is defined.	Required	63.0
recordType	String	Record type associated with the object type for which the underwriting rule group is defined.	Optional	63.0
rootProductId	String	ID of the root product.	Required	63.0
rulesetStatus	String	Status of the dynamic ruleset. Possible values are: <ul style="list-style-type: none"> Active Inactive Draft 	Required	63.0
stageTransitionId	String	ID of the stage transition.	Required	63.0
stageTransitionName	String	Name of the source stage transition.	Optional	63.0
toStage	String	Name of the stage the record is moving to.	Optional	63.0
underwritingRuleGroupId	String	ID of the underwriting rule group.	Optional	63.0

Insurance Underwriting Rule Group Update Status Input

Input representation of the request to update insurance underwriting rule group status.

JSON example

```
{
  "status": "Active"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
status	String	The status of the underwriting rule group.	Required	63.0

Insurance Update Underwriting Rules Input

Update Underwriting Rule Input Representation

JSON example

```
{
  "name": "Automatically Approve Quote",
  "apiName": "approveQuoteBasedonComprehensiveCoverageDeductibles",
  "effectiveFromDate": "2024-05-30T10:30:00",
  "effectiveToDate": "2024-11-01T10:30:00",
  "productPath": "01tDR000000EMslyAG/11BDR00000000T72AI", // "Auto Gold/Auto Vehicle"

  "ruleCriteria": [ // ruleCriteria will be replaced as a whole. Updating only
one condition is not supported
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Annual Milage",
          "attributePicklistValueId": null,
          "type": "Attribute",
          "attributeId": "0tjDR00000000VGyAY",
          "dataType": "Number",
          "values": [
            "50000"
          ]
        }
      ],
      "rootObjectId": "01tDR000000EMslyAG", // Auto Gold.
      "productRelatedComponent" : // Need More details on it.
      "criteriaIndex": 1,
      "sourceContextTagName": "ProductBasedOn",
      "sourceOperator": "Equals",
      "sourceDataType": "String",
      "sourceValues": [
        "11BDR00000000T72AI" // Auto Vehicle.
      ]
    },
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Comprehensive Deductible",
          "type": "Attribute",
          "attributePicklistValueId": null,
          "attributeId": "0tjDR00000000UaYAI",
          "dataType": "Number",
          "values": [
            "$5000"
          ]
        }
      ]
    }
  ]
}
```

```

    }
  ],
  "rootObjectId": "01tDR000000EMslYAG", // Auto Bundle
  "productRelatedComponent" : // Need More details on it.
  "criteriaIndex": 2,
  "sourceContextTagName": "Product",
  "sourceOperator": "Equals",
  "sourceDataType": "String",
  "sourceValues": [
    "01tDR000000EMsoYAG" // Comprehensive Product.
  ]
}
],
// EvaluationSuccessTaskGroup and EvaluationFailureTaskGroup are optional
// Fulfillment plan design time entities will be created.
"EvaluationSuccessTaskGroup": // Create FulfillmentStepDefinitionGroup records
{
  "name": "QuoteApproveSuccessTaskGroup",
  "tasks": [ // create FulfillmentStepDefinition records
    {
      "name": "KYC check flow", // FulfillmentStepDefinition API Names
      "type": "AutoTask", // FulfillmentStepDefinition type
      "flowApiName": "KYC check flow", // Flow API Names
    }
  ],
},
"EvaluationFailureTaskGroup": //Similar to EvaluationSuccessTaskGroup
{
  "name": "QuoteApproveFailureTaskGroup",
  "tasks": [
    {
      "name": "Vehicle ineligible",
      "type": "AutoTask",
      "flowApiName": "Mail Notification - Vehicle ineligible",
    }
  ]
}
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
description	String	Description of the underwriting rule.	Optional	63.0
effectiveFromDate	String	Date from which the underwriting rule is effective.	Optional	63.0
effectiveToDate	String	Date until which the underwriting rule is effective.	Optional	63.0
evaluationFailureTaskGroup	Evaluation Task Group Input[]	Input details for failed evaluation tasks.	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
evaluationSuccessfulGroup	Evaluation Task Group Input[]	Input details for successful evaluation tasks.	Optional	63.0
name	String	Name of the underwriting rule.	Optional	63.0
productPath	String	Name of the product path.	Optional	63.0
ruleCriteria	Insurance Rule Criteria Input[]	Rule criteria for the underwriting rule.	Required	63.0
status	String	Status of the underwriting rule. Possible values are: <ul style="list-style-type: none"> • Active • Inactive • Draft 	Optional	63.0

Responses

Learn more about the available response bodies of Insurance Product Administration APIs.

[Insurance Create Underwriting Rules](#)

Output representation details of insurance create underwriting rules request.

[Insurance Underwriting Rule Group Update Status](#)

Output representation of insurance underwriting rule group update status request.

[Insurance Underwriting Rules Error](#)

Details of the error response.

[Insurance Update Underwriting Rules](#)

Output representation details of the insurance update underwriting rules request.

Insurance Create Underwriting Rules

Output representation details of insurance create underwriting rules request.

JSON example

This example shows a sample success response.

```
{
  "underwritingRuleGroupId": "u8ixx0000007MnhEAI",
  "underwritingRuleId": "11Oxx0000007MnhEAE",
  "apiName": "ExampleApiName",
  "isSuccess": true
}
```


Property Name	Type	Description	Filter Group and Version	Available Version
apiName	String	API name of the underwriting rule.	Big, 63.0	63.0
errors	Insurance Underwriting Rules Error[]	List of errors encountered during the processing of the API request.	Big, 63.0	63.0
isSuccess	Boolean	Indicates whether the request is successful (<code>true</code>) or not (<code>false</code>).	Big, 63.0	63.0
underwritingRuleGroupId	String	ID of the underwriting rule group.	Big, 63.0	63.0
underwritingRuleId	String	ID of the underwriting rule.	Big, 63.0	63.0

Insurance Underwriting Rule Group Update Status

Output representation of insurance underwriting rule group update status request.

JSON example

```
{
  "success": "true",
  "underwritingRuleGroupId": "1KQxx0000004C92GAE"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
success	Boolean	Indicates whether the API request is successful (<code>true</code>) or not (<code>false</code>).	Big, 63.0	63.0
underwritingRuleGroupId	String	ID of the underwriting rule group whose ruleset status is updated.	Big, 63.0	63.0

Insurance Underwriting Rules Error

Details of the error response.

Property Name	Type	Description	Filter Group and Version	Available Version
code	String	Error code.	Big, 63.0	63.0
message	String	Message stating the reason for error, if any.	Big, 63.0	63.0

Insurance Update Underwriting Rules

Output representation details of the insurance update underwriting rules request.

JSON example

This example shows a sample success response.

```
{
  "underwritingRuleId": "11pxx0000005RE4AAM",
  "isActive": false,
  "isSuccess": true
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Insurance Underwriting Rules Error[]	List of errors encountered during the processing of the API request.	Big, 63.0	63.0
isActive	Boolean	Indicates whether the rule is active (<code>true</code>) or not (<code>false</code>).	Big, 63.0	63.0
isSuccess	Boolean	Indicates whether the request is successful (<code>true</code>) or not (<code>false</code>).	Big, 63.0	63.0
underwritingRuleId	String	ID of the underwriting rule.	Big, 63.0	63.0

CHAPTER 3 Insurance Quoting

In this chapter ...

- [Insurance Quoting Business APIs](#)
- [Insurance Quoting Standard Invocable Actions](#)
- [Insurance Quoting Developer Journey](#)

Use the Insurance Quoting developer resources to create, update, configure, and delete quotes and quote line items.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise, Unlimited,** and **Developer** Editions

Insurance Quoting Business APIs

Create, update, configure, and delete quotes and quote line items. Use the APIs to create guided flows for your external end users. This table lists the available Insurance Quoting resources.

Resource	Description
/connect/insurance/quotes (POST)	Generate a quote or quote-ready context in multiple ways.
/connect/insurance/product-rating (POST, PATCH)	Rate products or update product ratings for use in Insurance quoting processes.
/connect/insurance/product-surcharges/productSurchargeId (PATCH)	Update or fetch a product surcharge for a root product.
/connect/insurance/product-surcharges/bulk (POST)	Create one or more product surcharges to associate with a root product.
/connect/insurance/quotes/quoteId (GET, PATCH)	Update or get details for a specified quote ID.
/connect/insurance/underwriting-rules/invoke (POST)	Invoke underwriting rules for different object records during run time.

[Resources](#)

Learn more about the available Insurance Quoting API resources.

[Requests](#)

Learn more about the available Insurance Quoting requests.

[Responses](#)

Learn more about the available response bodies of Insurance Quoting APIs.

Resources

Learn more about the available Insurance Quoting API resources.

[Insurance Quote \(POST\)](#)

Generate a quote or quote-ready context in multiple ways. The API accepts user inputs, generates context, runs rules and pricing, prorates, and then returns a quote ID. The API can also use a context ID to run pricing and rules, update the context, and then return a quote ID. Alternatively, the API can accept user inputs, generate context, run rules and pricing, prorate, and then return only the context ID.

[Insurance Product Rating \(POST, PATCH\)](#)

Rate products or update product ratings for use in Insurance quoting processes.

[Insurance Product Surcharge \(PATCH\)](#)

Update or fetch a product surcharge for a root product.

[Insurance Product Surcharge Bulk \(POST\)](#)

Create one or more product surcharges to associate with a root product.

[Insurance Quote Details \(GET, PATCH\)](#)

Update or get details of a specified quote ID.

[Insurance Invoke Underwriting Rule \(POST\)](#)

Invoke underwriting rules for different object records during run time.

Insurance Quote (POST)

Generate a quote or quote-ready context in multiple ways. The API accepts user inputs, generates context, runs rules and pricing, prorates, and then returns a quote ID. The API can also use a context ID to run pricing and rules, update the context, and then return a quote ID. Alternatively, the API can accept user inputs, generate context, run rules and pricing, prorate, and then return only the context ID.

Resource

```
/connect/insurance/quotes
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/quotes
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "quoteName": "Auto Insurance Quote",
  "effectiveDate": "2024-06-30T00:00:00.000Z",
  "expirationDate": "2025-07-28T00:00:00.000Z",
  "opportunityId": "006xx000001a2odAAA",
  "transactionType": "AutoTransactionType",
  "salesTransactionTypeId": "1Chxx0000004C92CAE",
  "priceBookId": "01sxx0000005ptpAAA",
  "configOptions": {
    "executePricing": false,
    "validateProductCatalog": false,
    "executeConfigurationRules": false,
    "executeQualificationRules": false
  },
  "options": {
    "saveQuote": true,
    "productSellingModel": "One Time"
  },
  "quoteInputs": [
    {
      "productCode": "autoRoot",
```

```

"instanceKeys": [
  "AutoRoot"
],
"items": [
  {
    "productCode": "autoBundle",
    "instanceKeys": [
      "AutoRoot",
      "AutoBundle1"
    ],
    "attributes": {
      "avMake": "BMW",
      "avValue": "20000",
      "avYear": "2020",
      "AutoValueFactor": 5,
      "AutoAgeFactor": 6
    }
  },
  {
    "productCode": "autoDriver",
    "instanceKeys": [
      "AutoRoot",
      "AutoBundle1",
      "AutoDriver"
    ],
    "relatedRecords": [
      {
        "relatedRecordId": "001xx000003GYiEAAW",
        "relatedRecordObjName": "Account"
      },
      {
        "relatedRecordId": "001xx000003GYiEAAV",
        "relatedRecordObjName": "Contact"
      }
    ]
  },
  {
    "productCode": "autoComp",
    "instanceKeys": [
      "AutoRoot",
      "AutoBundle1",
      "AutoComp"
    ],
    "attributes": {
      "CompBasePremium": "750"
    }
  },
  {
    "productCode": "autoBundle",
    "instanceKeys": [
      "AutoRoot",
      "AutoBundle2"
    ],
    "attributes": {

```

```

        "avMake": "Audi",
        "avValue": "20000",
        "avYear": "2020",
        "AutoValueFactor": 5,
        "AutoAgeFactor": 6
    }
},
{
    "productCode": "autoDriver",
    "instanceKeys": [
        "AutoRoot",
        "AutoBundle2",
        "AutoDriver"
    ],
    "relatedRecords": [
        {
            "relatedRecordId": "001xx000003GYiEAAW",
            "relatedRecordObjName": "Account"
        },
        {
            "relatedRecordId": "001xx000003GYiEAAV",
            "relatedRecordObjName": "Contact"
        }
    ]
},
{
    "productCode": "autoCollision",
    "instanceKeys": [
        "AutoRoot",
        "AutoBundle2",
        "AutoCollision1"
    ],
    "attributes": {
        "CollisionBasePremium": "120"
    }
}
]
}
]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
accountId	String	ID of the account associated with the quote that's being created.	Required	63.0
additionalFields	Map<String, Object>	Additional fields that are used to create insurance quote.	Optional	63.0
configOptions	Insurance Quote Config Options []	The JSON structure that includes the options for configuration.	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
contextId	String	ID of the context that's used for creating the quote.	Required	63.0
effectiveDate	String	Date that the quote is effective from.	Required	63.0
expirationDate	String	Expiration date of the quote.	Optional	63.0
opportunityId	String	ID of the opportunity associated with the quote.	Optional	63.0
options	Insurance Quote Options[]	Insurance Quote Options.	Optional	63.0
priceBookId	String	ID of the pricebook associated with the quote.	Optional	63.0
pricingProcedure	String	Pricing Procedure for insurance quote.	Optional	63.0
quoteInputs	Insurance Quote Input Node[]	The JSON structure that contains user inputs for the products for which the quote is being created.	Required	63.0
quoteName	String	Name of the quote.	Optional	63.0
ratingDate	String	Date of the rating quote.	Optional	63.0
salesTransactionId	String	ID of the sales transaction type.	Optional	63.0
transactionType	String	Type of the transaction.	Optional	63.0

Response body for POST[Insurance Quote](#)**Insurance Product Rating (POST, PATCH)**

Rate products or update product ratings for use in Insurance quoting processes.

Resource

```
/connect/insurance/product-rating
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/product-rating
```

Available version

63.0

HTTP methods

PATCH, POST

Request body for PATCH**JSON example**

```
{
  "contextId": "11jxx0000006i5gAAA",
  "productId": "01txx0000006i5gAAA",
  "ratingInputs": [{
    "instanceKeys": ["AutoRoot"],
    "productCode": "AUTOROOT",
    "attributes": {
      "Deductible": 1000
    }
  },
  {
    "instanceKeys": ["AutoRoot", "Audi"],
    "productCode": "AUTO",
    "attributes": {
      "Mileage": 20000
    }
  },
  {
    "instanceKeys": ["AutoRoot", "Audi", "Tom"],
    "productCode": "DRIVER",
    "attributes": {
      "DriverAge": 50,
      "DriverAccident": 0
    },
    "targetRecords": ["003xx000004WkvEAAS"]
  }
],
  "ratingOptions": {
    "pricingProcedure": "AutoRootEP"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
addedNodes	Rating Input Node []	List of added nodes.	Optional	63.0
additionalFields	Map<String, Object>	Map of additional fields that must be updated in the quote context.	Optional	63.0
contextId	String	Context ID.	Required	63.0
deletedNodes	Rating Input Node []	List of deleted nodes.	Optional	63.0
effectiveDate	String	Date used to find the active pricing procedure for pricing. The default value is the current date.	Optional	63.0
ratingDate	String	Date that's used to find the active pricing procedure.	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
ratingOptions	Rating Options Input[]	Options for rating.	Required	63.0
updatedNodes	Rating Input Node[]	List of updated nodes.	Optional	63.0

Response body for PATCH[Product Rating](#)**Request body for POST****JSON example**

```
{
  "contextId": "11jxx0000006i5gAAA",
  "productId": "01txx0000006i5gAAA",
  "ratingInputs": [{
    "instanceKeys": ["AutoRoot"],
    "productCode": "AUTOROOT",
    "attributes": {
      "Deductible": 1000
    }
  },
  {
    "instanceKeys": ["AutoRoot", "Audi"],
    "productCode": "AUTO",
    "attributes": {
      "Mileage": 20000
    }
  },
  {
    "instanceKeys": ["AutoRoot", "Audi", "Tom"],
    "productCode": "DRIVER",
    "attributes": {
      "DriverAge": 50,
      "DriverAccident": 0
    },
    "targetRecords": ["003xx000004WkvEAAS"]
  }],
  "ratingOptions": {
    "pricingProcedure": "AutoRootEP"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Map<String, Object>	Map of additional fields that must be updated in the quote context.	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
effectiveDate	String	Date used to find the active pricing procedure for pricing. The default value is the current date.	Optional	63.0
productId	String	ID of the root product.	Required	63.0
ratingDate	String	Date used to find the active pricing procedure.	Optional	63.0
ratingInputs	Rating Input Node []	The input JSON that contains the attribute values of each context node.	Required	63.0
ratingOptions	Rating Options Input []	The options for rating.	Required	63.0
salesTransactionId	String	ID of the sales transaction type.	Optional	63.0
transactionType	String	ID of the transaction type.	Optional	63.0

Response body for POST[Product Rating Output](#)

Insurance Product Surcharge (PATCH)

Update or fetch a product surcharge for a root product.

Resource`/connect/insurance/product-surcharges/productSurchargeId`**Resource example**`https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/product-surcharges/productSurchargeId`**Available version**

63.0

HTTP methods

PATCH

Request body for PATCH**JSON example**

```
{
  "name": "AutoVehicleTax",
  "effectiveFromDate": "2024-05-30T10:30:00",
  "effectiveToDate": "2024-11-01T10:30:00",
  "isProrationEligible": true,
  "isRefundEligible": true,
  "isActive": true,
  "productPath": "01tDR000000EMs1YAG/11BDR00000000T72AI", // "Auto Gold/Auto Vehicle"
```

```

    "ruleDefinition": [ // ruleDefinition will be replaced as a whole. Updating
only one condition is not supported
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Annual Milage",
          "attributePicklistValueId": null,
          "type": "Attribute",
          "attributeId": "0tjDR00000000VGYAY",
          "dataType": "Number",
          "values": [
            "50000"
          ]
        }
      ],
      "rootObjectId": "01tDR000000EMslYAG", // Auto Gold.
      "productRelatedComponent" : "",
      "criteriaIndex": 1,
      "sourceContextTagName": "ProductBasedOn",
      "sourceOperator": "Equals",
      "sourceDataType": "String",
      "sourceValues": [
        "11BDR00000000T72AI" // Auto Vehicle.
      ]
    },
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Comprehensive Deductible",
          "type": "Attribute",
          "attributePicklistValueId": null,
          "attributeId": "0tjDR00000000UaYAI",
          "dataType": "Number",
          "values": [
            "$5000"
          ]
        }
      ],
      "rootObjectId": "01tDR000000EMslYAG", // Auto Bundle
      "productRelatedComponent" : "",
      "criteriaIndex": 2,
      "sourceContextTagName": "Product",
      "sourceOperator": "Equals",
      "sourceDataType": "String",
      "sourceValues": [
        "01tDR000000EMsoYAG" // Comprehensive Product.
      ]
    }
  ]
}

```

```

    ]
  }

```

Properties

Name	Type	Description	Required or Optional	Available Version
effectiveFromDate	String	Effective from date of the product surcharge.	Optional	63.0
effectiveToDate	String	Effective to date of the product surcharge.	Optional	63.0
isActive	Boolean	Indicates whether the product surcharge is active (<code>true</code>) or not (<code>false</code>). Currently, only active product surcharges are supported.	Optional	63.0
isProrationAllowed	Boolean	Indicates whether the product surcharge is allowed for proration (<code>true</code>) or not (<code>false</code>).	Optional	63.0
isRefundAllowed	Boolean	Indicates whether the product surcharge is allowed for refund (<code>true</code>) or not (<code>false</code>).	Optional	63.0
name	String	Name of the product surcharge.	Optional	63.0
productPath	String	A path containing the product IDs that roll up to the parent product.	Optional	63.0
productRelatedComponentId	String	Product related component associated with the product surcharge.	Optional	63.0
ruleApiName	String	Rule API name associated with product surcharge. You can set the ruleApiName with the POST or PATCH method. Once set, this parameter can't be changed. If you set the ruleApiName in the request body, you must also set the ruleCriteria.	Optional	63.0
ruleCriteria		Rule criteria for the dynamic rule.	Optional	63.0

Response body for PATCH
[Product Surcharge Patch](#)
Insurance Product Surcharge Bulk (POST)

Create one or more product surcharges to associate with a root product.

Resource

```
/connect/insurance/product-surcharges/bulk
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/product-surcharges/bulk
```

Available version

63.0

HTTP methods

POST

Request body for POST**JSON example**

```
{
  "records": [
    {
      "name": "AutoVehicleTax",
      "surchargeId": "01cLO000000EMslOKL",
      "rootProductId": "01tDR000000EMslYAG",
      "productPath": "01tDR000000EMslYAG/11BDR00000000T72AI",
      "effectiveFromDate": "2024-05-30T10:30:00",
      "effectiveToDate": "2024-11-01T10:30:00",
      "isProrationEligible": true,
      "isRefundEligible": true,
      "isActive": true,
      "ruleApiName": "autoVehicleTaxRule",
      "ruleCriteria": [
        {
          "conditions": [
            {
              "contextTagName": "SalesTransactionItemAttribute",
              "operator": "LessThan",
              "conditionIndex": 1,
              "attributeName": "Annual Milage",
              "attributePicklistValueId": null,
              "type": "Attribute",
              "attributeId": "0tjDR00000000VGYAY",
              "dataType": "Number",
              "values": [
                "50000"
              ]
            }
          ],
          "rootObjectId": "01tDR000000EMslYAG",
          "productRelatedComponent": "",
          "criteriaIndex": 1,
          "sourceContextTagName": "ProductBasedOn",
          "sourceOperator": "Equals",
          "sourceDataType": "String",
          "sourceValues": [
            "11BDR00000000T72AI"
          ]
        },
        {
          "conditions": [
```

```

        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Comprehensive Deductible",
          "type": "Attribute",
          "attributePicklistValueId": null,
          "attributeId": "0tjDR00000000UaYAI",
          "dataType": "Number",
          "values": [
            "$5000"
          ]
        },
        {
          "rootObjectId": "01tDR000000EMslYAG",
          "productRelatedComponent": "",
          "criteriaIndex": 2,
          "sourceContextTagName": "Product",
          "sourceOperator": "Equals",
          "sourceDataType": "String",
          "sourceValues": [
            "01tDR000000EMsoYAG"
          ]
        }
      ]
    }
  ]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
records	Product Surcharge Post Input[]	List of the product surcharges.	Required	63.0

Response body for POST

[Product Surcharge List](#)

Insurance Quote Details (GET, PATCH)

Update for get details of a specified quote ID.

Resource

```
/connect/insurance/quotes/quoteId
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/quotes/0Q0DR0000000xPw0AI
```

Available version

63.0

HTTP methods

GET, PATCH

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
connectDynamicParameters	Map<String,String>	Map of connect dynamic parameters.	Optional	63.0

Response body for GET[Get Quote](#)**Request body for PATCH****JSON example**

```
{
  "configOptions": {
    "executePricing": false,
    "validateProductCatalog": false,
    "executeConfigurationRules": false,
    "executeQualificationRules": false
  },
  "additionalFields": {
    "Status": "Draft",
    "ValidFromDate": "2025-10-29T00:00:00.000Z"
  },
  "addedNodes": [
    {
      "productCode": "autoCollision",
      "instanceKeys": [
        "autoBundle",
        "autoCollision"
      ],
      "additionalFields": {
        "StartDate": "2025-09-29T00:00:00.000Z",
        "EndDate": "2025-10-29T00:00:00.000Z",
        "Discount": 40
      }
    }
  ],
  "updatedNodes": [
    {
      "instanceKeys": [
        "autoBundle"
      ],
      "productCode": "autoBundle",
      "attributes": {
        "avMake": "AudiQ14"
      },
      "additionalFields": {
        "StartDate": "2025-09-29T00:00:00.000Z",
        "EndDate": "2025-10-29T00:00:00.000Z",

```



```

    "Discount": 50
  }
}
]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
addedNodes	Insurance Quote Input Node[]	List of added nodes for insurance quote.	Optional	63.0
additionalFields	Map<String, Object>	Insurance quote additional fields.	Optional	63.0
configOptions	Insurance Quote Config Options[]	Configurator options for insurance quote.	Optional	63.0
deletedNodes	Insurance Quote Input Node[]	List of deleted nodes from insurance quote.	Optional	63.0
pricingProcedure	String	Pricing procedure for insurance quote.	Optional	63.0
updatedNodes	Insurance Quote Input Node[]	List of updated nodes on insurance quote.	Optional	63.0

Response body for PATCH

[Insurance Quote](#)

Insurance Invoke Underwriting Rule (POST)

Invoke underwriting rules for different object records during run time.

Resource

```
/connect/insurance/underwriting-rules/invoke
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/underwriting-rules/invoke
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```

{
  "objectId": "110xx0000007MnhEAE",

```

```
"toStage": "Approved"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
objectId	String	ID of the insurance object(Quote or Insurance Policy ID).	Required	63.0
toStage	String	Name of the stage the record is moving to.	Required	63.0

Response body for POST

[Insurance Invoke Underwriting Rule](#)

Requests

Learn more about the available Insurance Quoting requests.

[Insurance Quote Config Options](#)

Input representation of insurance quote line item related objects.

[Insurance Quote Options](#)

Input representation of insurance quote options.

[Insurance Quote Related Record Input](#)

Input representation of insurance quote line item related objects.

[Insurance Invoke Underwriting Rule Input](#)

Input representation for invoke underwriting rule operation.

[Insurance Quote Input Node](#)

Input representation of root quote line items and child quote line items of insurance quote.

[Insurance Quote Patch Input](#)

Input representation for insurance quote patch operation.

[Insurance Quote Post Input](#)

Input representation details to create insurance quote.

[Product Rating Patch Input](#)

Input representation of product rating update details.

[Product Rating Post Input](#)

Input representation of the product rating details.

[Product Surcharge Bulk Post Input](#)

Input representation of the details of the product surcharge records.

[Product Surcharge Patch Input](#)

Input representation for product surcharge patch request.

[Product Surcharge Post Input](#)

Input representation of product surcharge details.

[Rating Input Node](#)

Input representation of details of context node.

[Rating Options Input](#)

Input representation details of rating service options.

Insurance Quote Config Options

Input representation of insurance quote line item related objects.

JSON example

```
"configOptions": {
  "executePricing": false,
  "validateProductCatalog": false,
  "executeConfigurationRules": false,
  "executeQualificationRules": false
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
executeConfigurationRules	Boolean	Indicates whether to execute configuration rules as part of insurance quote.	Optional	63.0
executePricing	Boolean	Indicates whether to execute NGP Pricing as part of insurance quote.	Optional	63.0
executeQualificationRules	Boolean	Indicates whether to execute qualification rules as part of insurance quote.	Optional	63.0
validateProductCatalog	Boolean	Indicates whether to execute product catalog validation as part of insurance quote.	Optional	63.0

Insurance Quote Options

Input representation of insurance quote options.

JSON example

```
options: {
  "saveQuote": true,
  "productSellingModel": "One Time"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
productSellingModel	String	Determines the pricebookId for the products.	Required	63.0
saveQuote	Boolean	Indicates whether to save insurance quote or only generate context.	Required	63.0

Insurance Quote Related Record Input

Input representation of insurance quote line item related objects.

JSON example

```
relatedRecords": [
  {
    "relatedRecordId": "001xx000003GYiEAAW",
    "relatedRecordObjName": "Account"
  },
  {
    "relatedRecordId": "001xx000003GYiEAAV",
    "relatedRecordObjName": "Contact"
  }
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
relatedRecordId	String	ID of the related insurance quote record.	Required	63.0
relatedRecordId relatedRecordObjName	String	Object name of the related insurance quote record.	Required	63.0

Insurance Invoke Underwriting Rule Input

Input representation for invoke underwriting rule operation.

JSON example

```
{
  "objectId": "110xx0000007MnhEAE",
  "toStage": "Approved"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
objectId	String	ID of the insurance object(Quote or Insurance Policy ID).	Required	63.0
toStage	String	Name of the stage the record is moving to.	Required	63.0

Insurance Quote Input Node

Input representation of root quote line items and child quote line items of insurance quote.

JSON example

```
quoteInputs: [
  {
    "productCode": "autoRoot",
    "instanceKeys": [
      "AutoRoot"
    ],
    "items": [
      {
        "productCode": "autoBundle",
        "instanceKeys": [
          "AutoRoot",
          "AutoBundle1"
        ],
        "attributes": {
          "avMake": "BMW",
          "avValue": "20000",
          "avYear": "2020",
          "AutoValueFactor": 5,
          "AutoAgeFactor": 6
        }
      },
      {
        "productCode": "autoDriver",
        "instanceKeys": [
          "AutoRoot",
          "AutoBundle1",
          "AutoDriver"
        ],
        "relatedRecords": [
          {
            "relatedRecordId": "001xx000003GYiEAAW",
            "relatedRecordObjName": "Account"
          },
          {
            "relatedRecordId": "001xx000003GYiEAAV",
            "relatedRecordObjName": "Contact"
          }
        ]
      }
    ]
  }
]
```

```

    ]
  },
  {
    "productCode": "autoComp",
    "instanceKeys": [
      "AutoRoot",
      "AutoBundle1",
      "AutoComp"
    ],
    "attributes": {
      "CompBasePremium": "750"
    }
  },
  {
    "productCode": "autoBundle",
    "instanceKeys": [
      "AutoRoot",
      "AutoBundle2"
    ],
    "attributes": {
      "avMake": "Audi",
      "avValue": "20000",
      "avYear": "2020",
      "AutoValueFactor": 5,
      "AutoAgeFactor": 6
    }
  },
  {
    "productCode": "autoDriver",
    "instanceKeys": [
      "AutoRoot",
      "AutoBundle2",
      "AutoDriver"
    ],
    "relatedRecords": [
      {
        "relatedRecordId": "001xx000003GYiEAAW",
        "relatedRecordObjName": "Account"
      },
      {
        "relatedRecordId": "001xx000003GYiEAAV",
        "relatedRecordObjName": "Contact"
      }
    ]
  },
  {
    "productCode": "autoCollision",
    "instanceKeys": [
      "AutoRoot",
      "AutoBundle2",
      "AutoCollision1"
    ],
    "attributes": {
      "CollisionBasePremium": "120"
    }
  }

```

```

    }
  }
]
}
]

```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Map<String, Object>	Insurance quote line items.	Optional	63.0
attributes	Map<String, Object>	Insurance quote attributes key-value pairs.	Optional	63.0
instanceKeys	String[]	Product instance identifier of the quote line item.	Required	63.0
items	String[]	The JSON structure that contains user inputs for the products for which the quote is created.	Optional	63.0
priceBookEntryId	String	ID of the pricebook associated with the quote.	Optional	63.0
productCode	String	Product code of the associated product.	Required	63.0
relatedRecords	Insurance Quote Related Record Input []	List of the related insurance quote records.	Required	62.0

Insurance Quote Patch Input

Input representation for insurance quote patch operation.

JSON example

```

{
  "configOptions": {
    "executePricing": false,
    "validateProductCatalog": false,
    "executeConfigurationRules": false,
    "executeQualificationRules": false
  },
  "additionalFields": {
    "Status": "Draft",
    "ValidFromDate": "2025-10-29T00:00:00.000Z"
  },
  "addedNodes": [
    {
      "productCode": "autoCollision",
      "instanceKeys": [
        "autoBundle",

```

```

        "autoCollision"
      ],
      "additionalFields": {
        "StartDate": "2025-09-29T00:00:00.000Z",
        "EndDate": "2025-10-29T00:00:00.000Z",
        "Discount": 40
      }
    },
    ],
    "updatedNodes": [
      {
        "instanceKeys": [
          "autoBundle"
        ],
        "productCode": "autoBundle",
        "attributes": {
          "avMake": "AudiQ14"
        },
        "additionalFields": {
          "StartDate": "2025-09-29T00:00:00.000Z",
          "EndDate": "2025-10-29T00:00:00.000Z",
          "Discount": 50
        }
      }
    ]
  }
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
addedNodes	Insurance Quote Input Node[]	List of added nodes for insurance quote.	Optional	63.0
additionalFields	Map<String, Object>	Insurance quote additional fields.	Optional	63.0
configOptions	Insurance Quote Config Options[]	Configurator options for insurance quote.	Optional	63.0
deletedNodes	Insurance Quote Input Node[]	List of deleted nodes from insurance quote.	Optional	63.0
pricingProcedure	String	Pricing procedure for insurance quote.	Optional	63.0
updatedNodes	Insurance Quote Input Node[]	List of updated nodes on insurance quote.	Optional	63.0

Insurance Quote Post Input

Input representation details to create insurance quote.

JSON example

```
{
  "quoteName": "Auto Insurance Quote",
  "effectiveDate": "2024-06-30T00:00:00.000Z",
  "expirationDate": "2025-07-28T00:00:00.000Z",
  "opportunityId": "006xx000001a2odAAA",
  "transactionType": "AutoTransactionType",
  "salesTransactionTypeId": "1Chxx0000004C92CAE",
  "priceBookId": "01sxx0000005ptpAAA",
  "configOptions": {
    "executePricing": false,
    "validateProductCatalog": false,
    "executeConfigurationRules": false,
    "executeQualificationRules": false
  },
  "options": {
    "saveQuote": true,
    "productSellingModel": "One Time"
  },
  "quoteInputs": [
    {
      "productCode": "autoRoot",
      "instanceKeys": [
        "AutoRoot"
      ],
      "items": [
        {
          "productCode": "autoBundle",
          "instanceKeys": [
            "AutoRoot",
            "AutoBundle1"
          ],
          "attributes": {
            "avMake": "BMW",
            "avValue": "20000",
            "avYear": "2020",
            "AutoValueFactor": 5,
            "AutoAgeFactor": 6
          }
        }
      ],
    },
    {
      "productCode": "autoDriver",
      "instanceKeys": [
        "AutoRoot",
        "AutoBundle1",
        "AutoDriver"
      ],
      "relatedRecords": [
        {
          "relatedRecordId": "001xx000003GYiEAAW",
          "relatedRecordObjName": "Account"
        },
        {

```

```

        "relatedRecordId": "001xx000003GYiEAAV",
        "relatedRecordObjName": "Contact"
    }
}
},
{
    "productCode": "autoComp",
    "instanceKeys": [
        "AutoRoot",
        "AutoBundle1",
        "AutoComp"
    ],
    "attributes": {
        "CompBasePremium": "750"
    }
},
{
    "productCode": "autoBundle",
    "instanceKeys": [
        "AutoRoot",
        "AutoBundle2"
    ],
    "attributes": {
        "avMake": "Audi",
        "avValue": "20000",
        "avYear": "2020",
        "AutoValueFactor": 5,
        "AutoAgeFactor": 6
    }
},
{
    "productCode": "autoDriver",
    "instanceKeys": [
        "AutoRoot",
        "AutoBundle2",
        "AutoDriver"
    ],
    "relatedRecords": [
        {
            "relatedRecordId": "001xx000003GYiEAAW",
            "relatedRecordObjName": "Account"
        },
        {
            "relatedRecordId": "001xx000003GYiEAAV",
            "relatedRecordObjName": "Contact"
        }
    ]
},
{
    "productCode": "autoCollision",
    "instanceKeys": [
        "AutoRoot",
        "AutoBundle2",
        "AutoCollision1"
    ]
}

```

```

    ],
    "attributes": {
      "CollisionBasePremium": "120"
    }
  }
]
}
]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
accountId	String	ID of the account associated with the quote that's being created.	Required	63.0
additionalFields	Map<String, Object>	Additional fields that are used to create insurance quote.	Optional	63.0
configOptions	Insurance Quote Config Options[]	The JSON structure that includes the options for configuration.	Optional	63.0
contextId	String	ID of the context that's used for creating the quote.	Required	63.0
effectiveDate	String	Date that the quote is effective from.	Required	63.0
expirationDate	String	Expiration date of the quote.	Optional	63.0
opportunityId	String	ID of the opportunity associated with the quote.	Optional	63.0
options	Insurance Quote Options[]	Insurance Quote Options.	Optional	63.0
priceBookId	String	ID of the pricebook associated with the quote.	Optional	63.0
pricingProcedure	String	Pricing Procedure for insurance quote.	Optional	63.0
quoteInputs	Insurance Quote Input Node[]	The JSON structure that contains user inputs for the products for which the quote is being created.	Required	63.0
quoteName	String	Name of the quote.	Optional	63.0
ratingDate	String	Date of the rating quote.	Optional	63.0
salesTransactionId	String	ID of the sales transaction type.	Optional	63.0
transactionType	String	Type of the transaction.	Optional	63.0

Product Rating Patch Input

Input representation of product rating update details.

JSON example

```
{
  "contextId": "11jxx0000006i5gAAA",
  "productId": "01txx0000006i5gAAA",
  "ratingInputs": [{
    "instanceKeys": ["AutoRoot"],
    "productCode": "AUTOROOT",
    "attributes": {
      "Deductible": 1000
    }
  },
  {
    "instanceKeys": ["AutoRoot", "Audi"],
    "productCode": "AUTO",
    "attributes": {
      "Mileage": 20000
    }
  },
  {
    "instanceKeys": ["AutoRoot", "Audi", "Tom"],
    "productCode": "DRIVER",
    "attributes": {
      "DriverAge": 50,
      "DriverAccident": 0
    },
    "targetRecords": ["003xx000004WkvEAAS"]
  }],
  "ratingOptions": {
    "pricingProcedure": "AutoRootEP"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
addedNodes	Rating Input Node []	List of added nodes.	Optional	63.0
additionalFields	Map<String, Object>	Map of additional fields that must be updated in the quote context.	Optional	63.0
contextId	String	Context ID.	Required	63.0
deletedNodes	Rating Input Node []	List of deleted nodes.	Optional	63.0
effectiveDate	String	Date used to find the active pricing procedure for pricing. The default value is the current date.	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
ratingDate	String	Date that's used to find the active pricing procedure.	Optional	63.0
ratingOptions	Rating Options Input[]	Options for rating.	Required	63.0
updatedNodes	Rating Input Node[]	List of updated nodes.	Optional	63.0

Product Rating Post Input

Input representation of the product rating details.

JSON example

```
{
  "contextId": "11jxx0000006i5gAAA",
  "productId": "01txx0000006i5gAAA",
  "ratingInputs": [{
    "instanceKeys": ["AutoRoot"],
    "productCode": "AUTOROOT",
    "attributes": {
      "Deductible": 1000
    }
  },
  {
    "instanceKeys": ["AutoRoot", "Audi"],
    "productCode": "AUTO",
    "attributes": {
      "Mileage": 20000
    }
  },
  {
    "instanceKeys": ["AutoRoot", "Audi", "Tom"],
    "productCode": "DRIVER",
    "attributes": {
      "DriverAge": 50,
      "DriverAccident": 0
    },
    "targetRecords": ["003xx000004WkvEAAS"]
  }],
  "ratingOptions": {
    "pricingProcedure": "AutoRootEP"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Map<String, Object>	Map of additional fields that must be updated in the quote context.	Optional	63.0
effectiveDate	String	Date used to find the active pricing procedure for pricing. The default value is the current date.	Optional	63.0
productId	String	ID of the root product.	Required	63.0
ratingDate	String	Date used to find the active pricing procedure.	Optional	63.0
ratingInputs	Rating Input Node []	The input JSON that contains the attribute values of each context node.	Required	63.0
ratingOptions	Rating Options Input []	The options for rating.	Required	63.0
salesTransactionType	String	ID of the sales transaction type.	Optional	63.0
transactionType	String	ID of the transaction type.	Optional	63.0

Product Surcharge Bulk Post Input

Input representation of the details of the product surcharge records.

JSON example

```
{
  "records": [
    {
      "name": "AutoVehicleTax",
      "surchargeId": "01cLO000000EMs1OKL",
      "rootProductId": "01tDR000000EMs1YAG",
      "productPath": "01tDR000000EMs1YAG/11BDR00000000T72AI",
      "effectiveFromDate": "2024-05-30T10:30:00",
      "effectiveToDate": "2024-11-01T10:30:00",
      "isProrationEligible": true,
      "isRefundEligible": true,
      "isActive": true,
      "ruleApiName": "autoVehicleTaxRule",
      "ruleCriteria": [
        {
          "conditions": [
            {
              "contextTagName": "SalesTransactionItemAttribute",
              "operator": "LessThan",
              "conditionIndex": 1,
              "attributeName": "Annual Milage",
              "attributePicklistValueId": null,

```

```

        "type": "Attribute",
        "attributeId": "0tjDR00000000VGYAY",
        "dataType": "Number",
        "values": [
            "50000"
        ]
    },
    ],
    "rootObjectId": "01tDR0000000EMslYAG",
    "productRelatedComponent": "",
    "criteriaIndex": 1,
    "sourceContextTagName": "ProductBasedOn",
    "sourceOperator": "Equals",
    "sourceDataType": "String",
    "sourceValues": [
        "11BDR00000000T72AI"
    ]
},
{
    "conditions": [
        {
            "contextTagName": "SalesTransactionItemAttribute",
            "operator": "LessThan",
            "conditionIndex": 1,
            "attributeName": "Comprehensive Deductible",
            "type": "Attribute",
            "attributePicklistValueId": null,
            "attributeId": "0tjDR00000000UaYAI",
            "dataType": "Number",
            "values": [
                "$5000"
            ]
        }
    ],
    "rootObjectId": "01tDR0000000EMslYAG",
    "productRelatedComponent": "",
    "criteriaIndex": 2,
    "sourceContextTagName": "Product",
    "sourceOperator": "Equals",
    "sourceDataType": "String",
    "sourceValues": [
        "01tDR0000000EMsoYAG"
    ]
}
]
}
]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
records	Product Surcharge Post Input[]	List of the product surcharges.	Required	63.0

Product Surcharge Patch Input

Input representation for product surcharge patch request.

JSON example

```
{
  "name": "AutoVehicleTax",
  "effectiveFromDate": "2024-05-30T10:30:00",
  "effectiveToDate": "2024-11-01T10:30:00",
  "isProrationEligible": true,
  "isRefundEligible": true,
  "isActive": true,
  "productPath": "01tDR000000EMslyAG/11BDR00000000T72AI", // "Auto Gold/Auto Vehicle"

  "ruleDefinition": [ // ruleDefinition will be replaced as a whole. Updating only
    one condition is not supported
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Annual Milage",
          "attributePicklistValueId": null,
          "type": "Attribute",
          "attributeId": "0tjDR00000000VGYAY",
          "dataType": "Number",
          "values": [
            "50000"
          ]
        }
      ],
      "rootObjectId": "01tDR000000EMslyAG", // Auto Gold.
      "productRelatedComponent" : "",
      "criteriaIndex": 1,
      "sourceContextTagName": "ProductBasedOn",
      "sourceOperator": "Equals",
      "sourceDataType": "String",
      "sourceValues": [
        "11BDR00000000T72AI" // Auto Vehicle.
      ]
    },
    {
      "conditions": [
        {
```



```

    "contextTagName": "SalesTransactionItemAttribute",
    "operator": "LessThan",
    "conditionIndex": 1,
    "attributeName": "Comprehensive Deductible",
    "type": "Attribute",
    "attributePicklistValueId": null,
    "attributeId": "0tjDR00000000UaYAI",
    "dataType": "Number",
    "values": [
      "$5000"
    ]
  },
  "rootObjectId": "01tDR000000EMsIYAG", // Auto Bundle
  "productRelatedComponent" : "",
  "criteriaIndex": 2,
  "sourceContextTagName": "Product",
  "sourceOperator": "Equals",
  "sourceDataType": "String",
  "sourceValues": [
    "01tDR000000EMsoYAG" // Comprehensive Product.
  ]
}
]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
effectiveFromDate	String	Effective from date of the product surcharge.	Optional	63.0
effectiveToDate	String	Effective to date of the product surcharge.	Optional	63.0
isActive	Boolean	Indicates whether the product surcharge is active (<code>true</code>) or not (<code>false</code>). Currently, only active product surcharges are supported.	Optional	63.0
isProrationAllowed	Boolean	Indicates whether the product surcharge is allowed for proration (<code>true</code>) or not (<code>false</code>).	Optional	63.0
isRefundAllowed	Boolean	Indicates whether the product surcharge is allowed for refund (<code>true</code>) or not (<code>false</code>).	Optional	63.0
name	String	Name of the product surcharge.	Optional	63.0
productPath	String	A path containing the product IDs that roll up to the parent product.	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
productRelatedComponentId	String	Product related component associated with the product surcharge.	Optional	63.0
ruleApiName	String	Rule API name associated with product surcharge. You can set the ruleApiName with the POST or PATCH method. Once set, this parameter can't be changed. If you set the ruleApiName in the request body, you must also set the ruleCriteria.	Optional	63.0
ruleCriteria		Rule criteria for the dynamic rule.	Optional	63.0

Product Surcharge Post Input

Input representation of product surcharge details.

JSON example

```
{
  "name": "AutoVehicleTax",
  "surchargeId": "01cLO000000EMslOKL",
  "rootProductId": "01tDR000000EMslYAG",
  "productPath": "01tDR000000EMslYAG/11BDR00000000T72AI",
  "effectiveFromDate": "2024-05-30T10:30:00",
  "effectiveToDate": "2024-11-01T10:30:00",
  "isProrationEligible": true,
  "isRefundEligible": true,
  "isActive": true,
  "ruleApiName": "autoVehicleTaxRule",
  "ruleCriteria": [
    {
      "conditions": [
        {
          "contextTagName": "SalesTransactionItemAttribute",
          "operator": "LessThan",
          "conditionIndex": 1,
          "attributeName": "Annual Milage",
          "attributePicklistValueId": null,
          "type": "Attribute",
          "attributeId": "0tjDR00000000VGYAY",
          "dataType": "Number",
          "values": [
            "50000"
          ]
        }
      ],
      "rootObjectId": "01tDR000000EMslYAG",
      "productRelatedComponent": "",
      "criteriaIndex": 1,
      "sourceContextTagName": "ProductBasedOn",
      "sourceOperator": "Equals",
    }
  ]
}
```

```

        "sourceDataType": "String",
        "sourceValues": [
            "11BDR00000000T72AI"
        ]
    }

```

Properties

Name	Type	Description	Required or Optional	Available Version
effectiveFromDate	String	Effective from date of the product surcharge.	Optional	63.0
effectiveToDate	String	Effective to date of the product surcharge.	Optional	63.0
isActive	Boolean	Indicates whether the product surcharge is active (<code>true</code>) or not (<code>false</code>). Currently, only active product surcharges are supported	Optional	63.0
isProrationAllowed	Boolean	Indicates whether the product surcharge is allowed for proration (<code>true</code>) or not (<code>false</code>).	Optional	63.0
isRefundAllowed	Boolean	Indicates whether the product surcharge is allowed for refund (<code>true</code>) or not (<code>false</code>).	Optional	63.0
name	String	Name of the product surcharge.	Required	63.0
productPath	String	A path containing the product IDs that roll up to the parent product.	Required	63.0
productRelatedComponentId	String	ID of the product related component associated with the product surcharge.	Optional	63.0
rootProductId	String	ID of the root product associated with the product surcharge.	Required	63.0
ruleApiName	String	Rule API name associated with product surcharge.	Optional	63.0
ruleCriteria		Rule criteria of the dynamic rule for the product surcharge.	Optional	63.0
surchargeId	String	ID of the surcharge associated with the product surcharge.	Required	63.0

Rating Input Node

Input representation of details of context node.

JSON example

```
ratingInputs": [{
  "instanceKeys": ["AutoRoot"],
  "productCode": "AUTOROOT",
  "attributes": {
    "Deductible": 1000
  }
},
{
  "instanceKeys": ["AutoRoot", "Audi"],
  "productCode": "AUTO",
  "attributes": {
    "Mileage": 20000
  }
},
{
  "instanceKeys": ["AutoRoot", "Audi", "Tom"],
  "productCode": "DRIVER",
  "attributes": {
    "DriverAge": 50,
    "DriverAccident": 0
  },
  "targetRecords": ["003xx000004WkvEAAS"]
}]
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Map<String, Object>	Map of additional fields that must be updated in the quote context.	Optional	63.0
attributes	Map<String, Object>	Map of attributes key-value pairs that's used to determine pricing.	Optional	63.0
instanceKeys	String[]	List of instance keys for context path.	Optional	63.0
productCode	String	Unique code to identify a product.	Optional	63.0
targetRecords	String[]	List of target records.	Optional	63.0

Rating Options Input

Input representation details of rating service options.

JSON example

```
ratingOptions": {
  "pricingProcedure": "AutoRootEP"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
<code>executeConfigurationRules</code>	Boolean	Indicates whether to execute configuration rules.	Optional	63.0
<code>executePricing</code>	Boolean	Indicates whether to execute pricing.	Optional	63.0
<code>pricingProcedure</code>	String	Name of the associated pricing procedure.	Optional	63.0

Responses

Learn more about the available response bodies of Insurance Quoting APIs.

[Get Quote](#)

Output representation of the details of the get quote details.

[Insurance Config Message](#)

Output representation of configurator messages.

[Insurance Invoke Underwriting Rule](#)

Output representation for invoke underwriting rule response.

[Insurance Quote](#)

Output representation of the details of the insurance quote.

[Insurance Underwriting Rule Result](#)

Underwriting rule evaluation result.

[Product Rating](#)

Output representation details of the product rating.

[Product Surcharge List](#)

Output Representation for List of ProductSurcharge

[Product Surcharge Patch](#)

Output representation of product surcharge update.

[Error Detail](#)

Output representation of the error details.

Get Quote

Output representation of the details of the get quote details.

JSON example

This example shows a sample success response.

```
{
  "contextId": "fa917d24acfaa40c14943d7009f096b2952c974a6e940d1302c465cd3abae3d",
  "errors": [],
  "insuranceContext": {
```

```

"SalesTransaction": [
  {
    "Status": "Draft",
    "Account": null,
    "BillingCity": null,
    "Subtotal": 1350,
    "businessObjectType": "Quote",
    "SalesTransactionName": "Multi Auto",
    "TotalAmount": 1350,
    "ShippingStreet": null,
    "SalesTransactionItem": [
      {
        "ProrationPolicy": "0muxx000000004sAAA",
        "ProductName": "Medical Payments",
        "ProductCode": "STA-3",
        "ProductSellingModel": "0jPxx000000004sEAA",
        "SpecificationType": "Coverage",
        "TaxTreatment": "1ttxx000000004rAAA",
        "SalesTransactionItemParent": "0Q0xx0000004CE1CAM",
        "Quantity": 1,
        "Product": "01txx0000006i7KAAQ",
        "businessObjectType": "QuoteLineItem",
        "PricebookEntry": "0luxx0000008ybrAAA",
        "PricingTermCount": 1,
        "StartDate": "2023-10-19T00:00:00Z",
        "NetUnitPrice": 750,
        "UnitPrice": 750,
        "id": "0QLxx0000004CIpGAM"
      },
      {
        "ProrationPolicy": null,
        "ProductName": "Collision",
        "ProductCode": "STA-2",
        "ProductSellingModel": "0jPxx000000004tEAA",
        "SpecificationType": "Coverage",
        "TaxTreatment": "1ttxx000000004rAAA",
        "SalesTransactionItemParent": "0Q0xx0000004CE1CAM",
        "Quantity": 1,
        "Product": "01txx0000006i7JAAQ",
        "businessObjectType": "QuoteLineItem",
        "PricebookEntry": "0luxx0000008ybsAAA",
        "PricingTermCount": 1,
        "StartDate": null,
        "NetUnitPrice": 500,
        "UnitPrice": 500,
        "id": "0QLxx0000004CIqGAM"
      }
    ],
    "BillingCountry": null,
    "BillingStreet": null,
    "Pricebook": "01sxx0000005pyfAAA",
    "Opportunity": null,
    "ShippingPostalCode": null,
    "SalesTransactionSource": "0Q0xx0000004CE1CAM",
  }
]

```

```

        "ShippingCountry": null,
        "ShippingCity": null,
        "ShippingState": null,
        "BillingPostalCode": null,
        "id": "0Q0xx0000004CE1CAM",
        "Contract": null,
        "BillingState": null
    }
}
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
contextId	String	ID of the insurance context.	Small, 63.0	63.0
errors	Error Response	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
insuranceContext	Map<String, Object>	List of insurance context details.	Small, 63.0	63.0

Insurance Config Message

Output representation of configurator messages.

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	Insurance quote configurator message.	Small, 63.0	63.0
type	String	Type of message.	Small, 63.0	63.0

Insurance Invoke Underwriting Rule

Output representation for invoke underwriting rule response.

JSON example

This example shows a sample success response.

```

{
  "objectId": "11pxx0000005RE4AAM",
  "objectName": "Quote",
  "stageTransitionId": "110xx0000007MnhEAE",
  "fromStage": "Draft",
  "toStage": "Approved",
  "criteriaExpression": "1 AND 2",
  "isSuccess": false,
  "underwritingRuleEvaluationId": "1yOxx0000007MnhEAE",
  "ruleResult": [
    {
      "ruleApiName": "CheckDriverAgeRule",

```

```

    "isSuccess": false
  },
  {
    "ruleApiName": "CheckVehicleValueRule",
    "isSuccess": true
  }
]
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
evaluationCriteria	String	Evaluation criteria expression.	Small, 63.0	63.0
fromStage	String	The from stage.	Small, 63.0	63.0
isSuccess	Boolean	Indicates whether the API request is successful (<code>true</code>) or not (<code>false</code>).	Small, 63.0	63.0
objectId	String	ID of the quote or insurance policy.	Small, 63.0	63.0
objectName	String	Name of the object (Quote or Insurance Policy).	Small, 63.0	63.0
ruleResult	Insurance Underwriting Rule Result Output[]	List of underwriting rule results.	Small, 63.0	63.0
stageTransitionName	String	ID of the stage transition.	Small, 63.0	63.0
stageTransitionId	String	ID of the underwriting rule evaluation.	Small, 63.0	63.0
toStage	String	The to stage.	Small, 63.0	63.0

Insurance Quote

Output representation of the details of the insurance quote.

JSON example

```

{
  "quoteId" : "0Q0xx0000004E2mCAE",
  "contextId" : "98977d60d58a614485280fe430afded87290baa9a6ad6bfec889df75ef11008e"
  "errors" : [],
  "success" : true
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
configMessages	Insurance Config Message[]	List of insurance quote configurator messages.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
contextId	String	ID of the context.	Small, 63.0	63.0
errors	Error Response	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
quoteDetails	Map<String, Object>	Details of the insurance quote.	Small, 63.0	63.0
quoteId	String	ID of the quote.	Small, 63.0	63.0
success	Boolean	Indicates whether the API request is successful (<code>true</code>) or not (<code>false</code>).	Small, 63.0	63.0

Insurance Underwriting Rule Result

Underwriting rule evaluation result.

Property Name	Type	Description	Filter Group and Version	Available Version
isSuccess	Boolean	Indicates whether the API request is successful (<code>true</code>) or not (<code>false</code>).	Small, 63.0	63.0
ruleApiName	String	API name of the dynamic rule.	Small, 63.0	63.0

Product Rating

Output representation details of the product rating.

JSON example

This example shows a sample success response.

```
{
  "contextId": "11jxx0000006i5gAAA",
  "addedNodes": [{
    "instanceKeys": ["AutoRoot", "BMW"],
    "productCode": "AUTO",
    "attributes": {
      "Mileage": 20000
    }
  }],
  "updatedNodes": [{
    "instanceKeys": ["AutoRoot", "Audi"],
    "productCode": "AUTO",
    "attributes": {
      "Mileage": 21000
    }
  }],
  "deletedNodes": [{
    "instanceKeys": ["AutoRoot", "Audi", "Tom"]
  }]
```

```

    }},
    "ratingOptions": {
      "pricingProcedure": "AutoRootEP"
    }
  }
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
configMessages	Insurance Config Message Output[]	List of configurator messages.	Small, 63.0	63.0
contextId	String	ID of the context.	Small, 63.0	63.0
errors	Error Response	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
pricingResult	Map<String, Object>	The pricing result.	Small, 63.0	63.0

Product Surcharge List

Output Representation for List of ProductSurcharge

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Map<String, Error Response >	List of errors encountered during the processing of the API request.	Big, 63.0	63.0
isSuccess	Boolean	Indicates whether the API request is successful (<code>true</code>) or not (<code>false</code>).	Big, 63.0	63.0
records	String[]	Product surcharge records that are created.	Big, 63.0	63.0

Product Surcharge Patch

Output representation of product surcharge update.

JSON example

This request shows a sample success response.

```

{
  "errors": []
  "isSuccess": true
  "productSurchargeId": "1Xrxx000000003F"
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Detail Output	List of errors encountered during the processing of the API request.	Big, 63.0	63.0
isSuccess	Boolean	Indicates whether the API request is successful (<code>true</code>) or not (<code>false</code>).	Big, 63.0	63.0
productSurchargeId	String	ID of the product surcharge.	Big, 63.0	63.0

Error Detail

Output representation of the error details.

Property Name	Type	Description	Filter Group and Version	Available Version
code	String	Error code.	Big, 63.0	63.0
message	String	Message stating the reason for error, if any.	Big, 63.0	63.0

Insurance Quoting Standard Invocable Actions

Learn more about the standard invocable actions available with Insurance Quoting.

[Create Insurance Quote Action](#)

Create an Insurance quote by using a context ID or a set of user inputs that represent quote details.

[Get Insurance Quote Details Action](#)

Get the details of an insurance quote.

[Reprice Insurance Product Action](#)

Recalculate the price of Insurance products based on user inputs.

Create Insurance Quote Action

Create an Insurance quote by using a context ID or a set of user inputs that represent quote details.

This action is available in API version 63.0 and later.

Supported REST HTTP Methods

URI

```
/services/data/v64.0/actions/standard/createInsuranceQuote
```

Formats

JSON

HTTP Methods

GET, POST

Authentication

Bearer Token

Inputs

Input	Details
quoteName	Type String Description The name of the quote being created.
accountId	Type String Description The ID of the account associated with the quote being created.
effectiveDate	Type Date Description The date that the quote becomes effective.
opportunityId	Type String Description Optional. The ID of the opportunity associated with the quote.
contextId	Type String Description Optional. The context ID used to issue the quote.
quoteInputs	Type String Description Optional. The set of user inputs on products that represent the quote details.
quoteOptions	Type Apex Type Description Optional. The options available when creating the quote.
pricingProcedureApiName	Type String

Input	Details
	Description Optional. The DeveloperName of the pricing procedure used to price the product.
pricebookId	Type String Description Optional. The ID of the pricebook associated with the quote.
expirationDate	Type Date Description Optional. The expiration date of the quote.
ratingDate	Type Date Description Optional. The date to use to find the active pricing procedure.
additionalFields	Type String Description Optional. The map of additional fields that must be updated in the quote context.

Example

Sample Request

```
{
  "inputs": [
    {
      "quoteName": "Auto Insurance Quote",
      "priceBookId" : "01sxx0000005pyfAAA",
      "accountId": "001xx000003GYwrAAG",
      "effectiveDate": "2024-10-11",
      "opportunityId": "006xx000001a2tbAAA",
      "quoteOptions": {
        "productSellingModel": "One Time",
        "saveQuote": true
      },
    },
    "quoteInputs":
  ]
}
```

Get Insurance Quote Details Action

Get the details of an insurance quote.
This action is available in API version 63.0 and later.

Supported REST HTTP Methods

- URI**
`/services/data/v64.0/actions/standard/getInsuranceQuoteDetails`
- Formats**
JSON
- HTTP Methods**
GET, POST
- Authentication**
Bearer Token

Inputs

Input	Details
quoteId	Type String Description The ID of the insurance quote to get the details for.
filters	Type String Description Optional. The filters used to filter the insurance quote or line items associated with the insurance quote.

Example

Sample Request

```
{
  "inputs":
  [
    {
      "quoteId": "0Q0xx0000004CXECA2"
    }
  ]
}
```

```
]
}
```

Reprice Insurance Product Action

Recalculate the price of Insurance products based on user inputs.

This action is available in API version 63.0 and later.

Supported REST HTTP Methods

URI

/services/data/v64.0/actions/standard/repriceInsuranceProduct

Formats

JSON

HTTP Methods

GET, POST

Authentication

Bearer Token

Inputs

Input	Details
contextId	Type String Description The context ID of the Insurance quote.
pricingProcedureApiName	Type String Description Optional. The developer name of the pricing procedure used to price the product.
addedNodes	Type String Description Optional. The set of added product instances details.
updatedNodes	Type String Description Optional. The set of updated product instances details.

Input	Details
deletedNodes	<p>Type</p> <p>String</p> <p>Description</p> <p>Optional. The set of deleted product instances details.</p>
effectiveDate	<p>Type</p> <p>Date</p> <p>Description</p> <p>Optional. The date that the quote becomes effective.</p>
ratingDate	<p>Type</p> <p>Date</p> <p>Description</p> <p>Optional. The date to use to find the active pricing procedure.</p>
additionalFields	<p>Type</p> <p>String</p> <p>Description</p> <p>Optional. The map of additional fields that must be updated in the insurance quote context.</p>
executeConfigurationRules	<p>Type</p> <p>Boolean</p> <p>Description</p> <p>Optional. Indicates whether to execute configuration rules or not. The default value is <code>false</code>.</p>

Example

Sample Request

```
{
  "inputs": [
    {
      "contextId": "273e01fdfb6f9f21428869eb784ee756976afb0e9bdc49f2a360be397a10ca12",
      "pricingProcedureApiName": "InsuranceDefaultPricingProcedure",
      "effectiveDate": "2024-10-18",
      "addedNodes":
        "[({\"instances\": [{\"id\": \"1\", \"type\": \"Auto\", \"attributes\": {\"DriveType\": \"Auto\", \"tagCode\": \"03\"}}, {\"id\": \"2\", \"type\": \"Auto\", \"attributes\": {\"DriveType\": \"Auto\", \"tagCode\": \"03\"}}, {\"id\": \"3\", \"type\": \"Auto\", \"attributes\": {\"DriveType\": \"Auto\", \"tagCode\": \"03\"}}, {\"id\": \"4\", \"type\": \"Auto\", \"attributes\": {\"DriveType\": \"Auto\", \"tagCode\": \"03\"}}], \"tagCode\": \"03\"})]"
    }
  ]
}
```


Insurance Quoting Developer Journey

Learn how to use the Insurance Quoting and Rating APIs to build user journeys for creating and updating quotes with Salesforce Flows and Omniscripts. These APIs help you build user journeys for users, such as insurance representatives who use the Salesforce console as well as the policyholders and partners who use the services through community portals.

Refer to these use cases to deepen your knowledge about these user journeys.

[Build a Self-service Quote and Policy Creation Functionality for Community Portal Users](#)

This user journey applies when a user, such as a policyholder or partner, logs into a customer community portal to select a product, view its price, and configure its attributes. The user then creates a quote, and proceeds to issue a policy.

[Calculate Quote Price Without Creating a Formal Quote](#)

This user journey applies when a customer wants to see an estimated quote price based on their requirements without having to save a formal quote.

[Implement Quote Creation Functionality for Insurance Representatives](#)

This user journey applies when a policyholder contacts an insurance company and an insurance representative guides them through identifying coverage needs, configuring and pricing a quote, and sharing it for review.

[Implement Quote Update Functionality for Insurance Representatives](#)

This user journey applies when a policyholder wants to update an existing quote, such as adding, deleting, or editing quote line items.

Build a Self-service Quote and Policy Creation Functionality for Community Portal Users

This user journey applies when a user, such as a policyholder or partner, logs into a customer community portal to select a product, view its price, and configure its attributes. The user then creates a quote, and proceeds to issue a policy.

Assumption

It's assumed that the user logs into the community portal, selects a product to generate a quote, and then gets the policy issued.

Use these steps to build the user journey.

1. **Generate Priced Context**—Make a POST request on the [Insurance Product Rating](#) Connect API with the required inputs to generate the price adjusted context.
This request returns the context ID in the response.
2. **Create Quote**—Use the context ID to create a quote either by:
 - Making a POST request on the [Insurance Quote](#) Connect API, or
 - Using the [Create Insurance Quote](#) invocable action.
 This request returns a quote ID in the response.
3. **Get Quote Context**—Use the quote ID to retrieve the quote details either by:
 - Making a GET request on the [Insurance Quote Details](#) Connect API, or
 - Using the [Get Insurance Quote Details](#) invocable action.
 This request returns the context ID in the response that contains the context of the quote.
4. **Issue Policy**—Use the retrieved quote context to create a policy record either by:
 - Making a POST request on the [Issue Insurance Policy](#) Connect API, or

- Using the [Issue Insurance Policy](#) invocable action.

Calculate Quote Price Without Creating a Formal Quote

This user journey applies when a customer wants to see an estimated quote price based on their requirements without having to save a formal quote.

Use the Quoting API if Quoting is managed within Salesforce. Use the Product Rating API when Quoting is not managed by Salesforce or when a third-party system handles the quoting process.

1. Calculate the quote price by using one of these methods.
 - **Quoting API**—Make a POST request on the [Insurance Quote](#) Connect API. Set the `saveQuote` parameter to `false`.
 - **Product Rating API**—Make a POST request on the [Insurance Product Rating](#) Connect API to generate a price-adjusted context based on user inputs without creating a formal quote.

Implement Quote Creation Functionality for Insurance Representatives

This user journey applies when a policyholder contacts an insurance company and an insurance representative guides them through identifying coverage needs, configuring and pricing a quote, and sharing it for review.

Assumption

It's assumed that a valid quote ID is available to update the existing quote.

The insurance representative collects the policyholder's requirements and adds any other additional rules during the API call to create a quote.

1. **Create Quote**—Create a quote either by:
 - Making a POST request on the [Insurance Quote](#) Connect API, or
 - Using the [Create Insurance Quote](#) invocable action.

This request returns a quote ID in the response.
2. **(Optional) Invoke Underwriting Review**—To invoke an underwriting review after the initial quote is created, make a POST request on the [Insurance Invoke Underwriting Rule](#) Connect API with the necessary inputs.
3. Use the quote ID that's generated during the quote creation process for further processing, such as policy issuance.

Implement Quote Update Functionality for Insurance Representatives

This user journey applies when a policyholder wants to update an existing quote, such as adding, deleting, or editing quote line items.

Assumption

It's assumed that the insurance representative has collected the policyholder's requirements to update a quote.

Use the quote ID to update the quote.

1. Make a PATCH request on the [Insurance Quote Details](#) Connect API to update the quote.
This request returns a new price-adjusted quote ID in the response.

CHAPTER 4 Insurance Policy Administration

In this chapter ...

- [Insurance Policy Business APIs](#)
- [Insurance Policy Standard Invocable Actions](#)
- [Insurance Policy Administration Metadata API Types](#)
- [Insurance Policy Administration Apex Reference](#)
- [Insurance Policy Developer Journey](#)

Use the Insurance Policy developer resources to manage insurance policies comprehensively, for different transaction types, such as issuance, endorsement, renewal, and cancellation. Each transaction type creates an InsurancePolicyTransaction record.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise, Unlimited, and Developer** Editions

Insurance Policy Business APIs

Manage policy transactions such as issuance, endorsement, renewal, and cancellation. Calculate term taxes based on transaction types and policy durations.

This table lists the available Insurance Policy Administration resources.

Resource	Description
/connect/insurance/policies/policyId/cancel (POST)	Cancel an insurance policy version. The API takes the effective date of cancellation from the input JSON.
/connect/insurance/policies/policyId/endorse (POST)	Endorse the latest version of the insurance policy based on the context ID and effective date.
/connect/insurance/policies/policyId?_qs=contextOnly=true (GET)	Get the ID of the context and the insurance policy details. This includes related assets, participants, and coverages. Use query parameters to request additional information.
/connect/insurance/policies (POST)	Use the context ID provided in the input JSON to generate a new insurance policy.
/connect/insurance/policies/policyId/renew (POST)	Use the context ID of an existing policy to renew a policy. Modify the renewal date and also add optional fields and additional information in the input JSON.

[Resources](#)

Learn more about the available Insurance Policy API resources.

[Requests](#)

Learn more about the available Insurance Policy requests.

[Responses](#)

Learn more about the available response bodies of Insurance Policy APIs.

Resources

Learn more about the available Insurance Policy API resources.

[Cancel Insurance Policy \(POST\)](#)

Cancel an insurance policy version. The API takes the effective date of cancellation from the input JSON. You can advance the cancellation date within the policy term, but you can't postpone the cancellation.

[Endorse Insurance Policy \(POST\)](#)

Endorse the latest version of the insurance policy based on the context ID and effective date. You can also include other optional fields and additional details in the input JSON.

[Insurance Policy Details \(GET\)](#)

Get the context ID and the insurance policy details. This includes related assets, participants, and coverages. Use query parameters to request additional information.

[Issue Insurance Policy \(POST\)](#)

Use the context ID provided in the input JSON to generate a new insurance policy.

[Renew Insurance Policy \(POST\)](#)

Use the context ID of an existing policy to renew a policy. Modify the renewal date and also add optional fields and additional information in the input JSON.

Cancel Insurance Policy (POST)

Cancel an insurance policy version. The API takes the effective date of cancellation from the input JSON. You can advance the cancellation date within the policy term, but you can't postpone the cancellation.

Resource

```
/connect/insurance/policies/policyId/cancel
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YT5x0000000006TCAG/cancel
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "effectiveDate": "2024-09-08",
  "transactionRecord": {
    "name": "Cancel Transaction"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
effectiveDate	String	Date on which the cancellation of the policy goes into effect.	Required	63.0
transactionRecord	Insurance Policy Transaction Input[]	The JSON structure that overrides the transaction information.	Optional	63.0

Response body for POST

[Cancel Policy](#)

Endorse Insurance Policy (POST)

Endorse the latest version of the insurance policy based on the context ID and effective date. You can also include other optional fields and additional details in the input JSON.

Resource

```
/connect/insurance/policies/policyId/endorse
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTxx00000000WHGAY/endorse
```

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
insurancePolicy:{
  "policyName":"p2_endorse_name",
  "policyNumber":"p2_endorse_number",
  "effectiveDate": "2024-01-01"
},
"transaction":{
  "name":"New Endorsement"
},
"additionalFields":{
  "additionalFieldsList":[
    {
      "additionalFieldsMapValue":{
        "instanceKey":"autoRoot",
        "fields":{
          "Checkbox__c":"true",
          "Currency__c":123456,
          "Date__c":"2024-11-30",
          "DateTime__c":"2024-12-31T01:34:00.000Z",
```

```

        "Email__c": "testuser@sf.com"
      }
    }
  }

  {
    "additionalFieldsMapValue": {
      "instanceKey": "autoDriver",
      "fields": {
        "TextEncr__c": "233345675676",
        "Url__c": "https://thisistest.com",
        "Time__c": "12:05:00.000Z"
      }
    }
  }
]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Additional Fields Map Value[]	The JSON structure that's used to add custom or standard fields to the InsurancePolicy, InsurancePolicyAsset, InsurancePolicyCoverage, and InsurancePolicyParticipant objects. The values for additionalFields take precedence over the context or the insurancePolicy JSON. This is applicable only for the endorsed policy version. Address and Geolocation field types are not supported as additional fields.	Optional	63.0
contextId	String	Context ID of the quote that's used for endorsement.	Required	63.0
effectiveDate	String	Date from when endorsed policy version is effective from.	Required	63.0
insurancePolicy	Insurance Policy Input[]	The JSON structure that overrides the name and policy number of the endorsed policy version.	Optional	63.0
transactionRecord	Insurance Policy Transaction Input[]	The JSON structure that overrides the transaction information.	Optional	63.0

Response body for POST[Policy](#)

Insurance Policy Details (GET)

Get the context ID and the insurance policy details. This includes related assets, participants, and coverages. Use query parameters to request additional information.

Resource

```
/connect/insurance/policies/policyId?_qs=contextOnly=true
```

```
/connect/insurance/policies/policyId?_qs=expand[attributes]=true
```

```
/connect/insurance/policies/policyId?_qs=expand[attributes]=true&expand[surcharges]=true
```

```
/connect/insurance/policies/policyId?_qs=expand[additionalFields] [fieldSet] [$in]={fieldSetIds}
```

```
/connect/insurance/policies/policyId?_qs=expand[additionalFields] [fieldSet] [$eq]={fieldSetId}
```

```
/connect/insurance/policies/policyId?_qs=expand[additionalFields] [fieldSet] [$eq]={fieldSetId};contextOnly=true
```

```
/connect/insurance/policies/policyId?_qs=expand[additionalFields] [fieldSet] [$eq]={fieldSetId};expand[attributes]=true
```

```
/connect/insurance/policies/policyId?_qs=expand[additionalFields] [fieldSet] [$eq]={fieldSetId};expand[attributes]=true&expand[surcharges]=true
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTxx0000000001GAA
```

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTS0000000233R4AQ?_qs=expand[attributes]=true
```

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTx0000000001GAA?_qs=expand[attributes]=true&expand[surcharges]=true
```

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTS0000000233R4AQ?_qs=contextOnly=true
```

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTS0000000233R4AQ?_qs=expand[additionalFields] [fieldSet] [$in]=0XS00001A03AG,0XS00001A03AG
```

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTS0000000233R4AQ?_qs=expand[additionalFields] [fieldSet] [$in]=0XS00001A03AG,0XS00001A03AG;contextOnly=true
```

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTS0000000233R4AQ?_qs=expand[additionalFields] [fieldSet] [$in]=0XS00001A03AG,0XS00001A03AG;expand[attributes]=true
```

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTx0000000001GAA?_qs=expand[surcharges]=true
```

Available version

63.0

HTTP methods

GET

Request parameters for GET

Parameter Name	Type	Description	Required or Optional	Available Version
connectDynamicParameters	Map<String,String>	Map of connect dynamic parameters.	Optional	63.0

Response body for GET

[Get Policy](#)

Issue Insurance Policy (POST)

Use the context ID provided in the input JSON to generate a new insurance policy.

Resource

/connect/insurance/policies

Resource example

https://**yourInstance**.salesforce.com/services/data/v64.0/connect/insurance/policies

Available version

63.0

HTTP methods

POST

Request body for POST

JSON example

```
{
  "contextId": "3561319f267fd53380bc38c049a01c8089876e833a321860766d35a9c40b67df",
  "insurancePolicy": {
    "policyName": "Mr. Bret Lees Policy",
    "policyNumber": "Policy-Bret-OneYear",
    "effectiveFromDate": "2024-01-01",
    "effectiveToDate": "2024-10-31"
  },
  "transaction": {
    "name": <transcation_name>
  },
  "additionalFields":{
    "additionalFieldsList":[
      {
        "additionalFieldsMapValue":{
          "instanceKey":"autoRoot",
          "fields":{
            "Checkbox__c":"true",
            "Currency__c":123456,
            "Date__c":"2024-11-30",
            "DateTime__c":"2024-12-31T01:34:00.000Z",
            "Email__c":"testuser@sf.com"
          }
        }
      }
    ]
  }
}
```

```

    }
  }
}

{
  "additionalFieldsMapValue":{
    "instanceKey":"autoDriver",
    "fields":{
      "TextEncr__c":"233345675676",
      "Url__c":"https://thisistest.com",
      "Time__c":"12:05:00.000Z"
    }
  }
}
]
}
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Additional Fields Map Value[]	The JSON structure that's used to add custom or standard fields to the InsurancePolicy, InsurancePolicyAsset, InsurancePolicyCoverage, and InsurancePolicyParticipant objects. The values for additionalFields take precedence over the context or the insurancePolicy JSON. Address and Geolocation field types are not supported as additional fields. Add the SourceQuoteld standard field to the additionalFields option when the context ID in the input JSON for the Issue Insurance Policy API is generated by the Create Insurance Quote API.	Optional	63.0
contextId	String	Context ID of the quote that's used to create the policy.	Required	63.0
insurancePolicy	Insurance Policy Input[]	The JSON structure that's used to create the policy. It contains fields and values from the Insurance Policy object.	Optional	63.0
transactionRecord	Insurance Policy Transaction Input[]	The JSON structure that's used to override transaction information.	Optional	63.0

Response body for POST[Policy](#)

Renew Insurance Policy (POST)

Use the context ID of an existing policy to renew a policy. Modify the renewal date and also add optional fields and additional information in the input JSON.

Resource

```
/connect/insurance/policies/policyId/renew
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/policies/0YTx00000000WHGAY/renew
```

Available version

63.0

HTTP methods

POST

Request body for POST**JSON example**

```
{
  "contextId": "54920265842e1b61496c38b7e2202f6f0aff8123ad2f19fe17ac68f7fc1c60b3",
  "insurancePolicy": {
    "policyName": "Demo Test Policy 9",
    "policyNumber": "test renew policy",
    "effectiveFromDate": "2025-03-15",
    "effectiveToDate": "2026-03-15"
  },
  "transactionRecord": {
    "name": "Renew Payment"
  },
  "additionalFields": {
    "additionalFieldsList": [
      {
        "additionalFieldsMapValue": {
          "instanceKey": "autoRoot",
          "fields": {
            "Checkbox__c": "true",
            "Currency__c": 123456,
            "Date__c": "2024-11-30",
            "DateTime__c": "2024-12-31T01:34:00.000Z",
            "Email__c": "testuser@sf.com"
          }
        }
      }
    ]
  },
  {
    "additionalFieldsMapValue": {
      "instanceKey": "autoDriver",
      "fields": {
```

```
      "TextEncr__c": "233345675676",
      "Url__c": "https://thisistest.com",
      "Time__c": "12:05:00.000Z"
    }
  }
}
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Additional Fields Map Value[]	The JSON structure that's used to add custom or standard fields to the InsurancePolicy, InsurancePolicyAsset, InsurancePolicyCoverage, and InsurancePolicyParticipant objects. The values for additionalFields take precedence over the context or the insurancePolicy JSON. This is applicable only for the endorsed policy version. Address and Geolocation field types are not supported as additional fields.	Optional	63.0
contextId	String	Context ID of the quote that's used to renew the policy.	Required	63.0
insurancePolicy	Insurance Policy Input[]	The JSON structure that's used to renew the policy. It contains fields and values from the Insurance Policy object.	Optional	63.0
transactionRecord	Insurance Policy Transaction Input[]	The JSON structure that's used to override the transaction information.	Optional	63.0

Response body for POST
[Policy](#)

Requests

Learn more about the available Insurance Policy requests.

[Additional Fields Map Value](#)

AdditionalFieldsMapValue wrapper object for use as child wrapper of AdditionalFieldsMap.

[Cancel Policy Input](#)

Input Representation of cancel policy details.

[Endorse Policy Input](#)

Input Representation of endorse policy details.

[Insurance Policy Input](#)

Input representation of insurance policy details.

[Insurance Policy Transaction Input](#)

Input representation of the insurance policy transaction details.

[Issue Policy Input](#)

Input representation of the request to issue an insurance policy.

Additional Fields Map Value

AdditionalFieldsMapValue wrapper object for use as child wrapper of AdditionalFieldsMap.

JSON example

```
additionalFields":{
  "additionalFieldsList":[
    {
      "additionalFieldsMapValue":{
        "instanceKey":"autoRoot",
        "fields":{
          "Checkbox__c":"true",
          "Currency__c":123456,
          "Date__c":"2024-11-30",
          "DateTime__c":"2024-12-31T01:34:00.000Z",
          "Email__c":"testuser@sf.com"
        }
      }
    }
  ],
  {
    "additionalFieldsMapValue":{
      "instanceKey":"autoDriver",
      "fields":{
        "TextEncr__c":"233345675676",
        "Url__c":"https://thisistest.com",
        "Time__c":"12:05:00.000Z"
      }
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFieldsMapValue	Object	The JSON map that contains the fields to be updated.	Required	63.0

Cancel Policy Input

Input Representation of cancel policy details.

JSON example

```
{
  "effectiveDate": "2024-09-08",
  "transactionRecord": {
    "name" : "Cancel Transaction"
  }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
effectiveDate	String	Date on which the cancellation of the policy goes into effect.	Required	63.0
transactionRecord	Insurance Policy Transaction Input[]	The JSON structure that overrides the transaction information.	Optional	63.0

Endorse Policy Input

Input Representation of endorse policy details.

JSON example

```
insurancePolicy:{
  "policyName":"p2_endorse_name",
  "policyNumber":"p2_endorse_number",
  "effectiveDate": "2024-01-01"
},
"transaction":{
  "name":"New Endorsement"
},
"additionalFields":{
  "additionalFieldsList":[
    {
      "additionalFieldsMapValue":{
        "instanceKey":"autoRoot",
        "fields":{
          "Checkbox__c":"true",
          "Currency__c":123456,
          "Date__c":"2024-11-30",
          "DateTime__c":"2024-12-31T01:34:00.000Z",
          "Email__c":"testuser@sf.com"
        }
      }
    }
  ]
}

{
  "additionalFieldsMapValue":{
```

```

        "instanceKey": "autoDriver",
        "fields": {
            "TextEncr__c": "233345675676",
            "Url__c": "https://thisistest.com",
            "Time__c": "12:05:00.000Z"
        }
    }
}
]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Additional Fields Map Value[]	The JSON structure that's used to add custom or standard fields to the InsurancePolicy, InsurancePolicyAsset, InsurancePolicyCoverage, and InsurancePolicyParticipant objects. The values for additionalFields take precedence over the context or the insurancePolicy JSON. This is applicable only for the endorsed policy version. Address and Geolocation field types are not supported as additional fields.	Optional	63.0
contextId	String	Context ID of the quote that's used for endorsement.	Required	63.0
effectiveDate	String	Date from when endorsed policy version is effective from.	Required	63.0
insurancePolicy	Insurance Policy Input[]	The JSON structure that overrides the name and policy number of the endorsed policy version.	Optional	63.0
transactionRecord	Insurance Policy Transaction Input[]	The JSON structure that overrides the transaction information.	Optional	63.0

Insurance Policy Input

Input representation of insurance policy details.

JSON example

```

insurancePolicy": {
    "policyName": "Mr. Bret Lees Policy",
    "policyNumber": "Policy-Bret-OneYear",

```

```
"effectiveFromDate": "2024-01-01",
"effectiveToDate": "2024-10-31"
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
effectiveFromDate	String	Date that the policy is effective from.	Optional	63.0
effectiveToDate	String	Expiration date of the policy.	Optional	63.0
policyName	String	Name of the insurance policy.	Optional	63.0
policyNumber	String	Number of the policy.	Optional	63.0

Insurance Policy Transaction Input

Input representation of the insurance policy transaction details.

JSON example

```
transaction": {
  "name": Cancel Transaction
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
name	String	Name of the transaction, such as Cancel Transaction.	Optional	63.0

Issue Policy Input

Input representation of the request to issue an insurance policy.

JSON example

```
{
  "contextId": "3561319f267fd53380bc38c049a01c8089876e833a321860766d35a9c40b67df",
  "insurancePolicy": {
    "policyName": "Mr. Bret Lees Policy",
    "policyNumber": "Policy-Bret-OneYear",
    "effectiveFromDate": "2024-01-01",
    "effectiveToDate": "2024-10-31"
  },
  "transaction": {
    "name": <transcation_name>
  },
  "additionalFields":{
    "additionalFieldsList":[
```



```
{
  "additionalFieldsMapValue": {
    "instanceKey": "autoRoot",
    "fields": {
      "Checkbox__c": "true",
      "Currency__c": 123456,
      "Date__c": "2024-11-30",
      "DateTime__c": "2024-12-31T01:34:00.000Z",
      "Email__c": "testuser@sf.com"
    }
  }
}

{
  "additionalFieldsMapValue": {
    "instanceKey": "autoDriver",
    "fields": {
      "TextEncr__c": "233345675676",
      "Url__c": "https://thisistest.com",
      "Time__c": "12:05:00.000Z"
    }
  }
}
]
```

Properties

Name	Type	Description	Required or Optional	Available Version
additionalFields	Additional Fields Map Value[]	The JSON structure that's used to add custom or standard fields to the InsurancePolicy, InsurancePolicyAsset, InsurancePolicyCoverage, and InsurancePolicyParticipant objects. The values for additionalFields take precedence over the context or the insurancePolicy JSON. Address and Geolocation field types are not supported as additional fields. Add the SourceQuoteld standard field to the additionalFields option when the context ID in the input JSON for the Issue Insurance Policy API is generated by the Create Insurance Quote API.	Optional	63.0
contextId	String	Context ID of the quote that's used to create the policy.	Required	63.0

Name	Type	Description	Required or Optional	Available Version
insurancePolicy	Insurance Policy Input[]	The JSON structure that's used to create the policy. It contains fields and values from the Insurance Policy object.	Optional	63.0
transactionRecord	Insurance Policy Transaction Input[]	The JSON structure that's used to override transaction information.	Optional	63.0

Responses

Learn more about the available response bodies of Insurance Policy APIs.

[Cancel Policy](#)

Output representation of cancel policy details.

[Get Policy](#)

Output representation of policy details.

[Policy](#)

Output representation of policy details.

[Refund](#)

Output representation of refund details.

Cancel Policy

Output representation of cancel policy details.

JSON example

```
{
  "impactedPolicyVersions": [
    "0YTxx000000006TGAQ"
  ],
  "refund": {
    "fee": 15.71,
    "premium": 31.42,
    "tax": 15.71
  }
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
error	Error Response	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
impactedPolicyVersions	String[]	List of impacted policy versions.	Small, 63.0	63.0
refund	Refund[]	Refund associated with cancellation operation.	Small, 63.0	63.0

Get Policy

Output representation of policy details.

JSON example

This example shows a sample response when you pass a policy ID in the get policy request.

```
{
  "insurancepolicy" : {
    "id" : "xxxxxxxxXxx",
    "policyName" : "policyName",
    "policyNumber" : "PN-10012",
    "universalPolicyNumber" : "TNS-123981",
    "effectiveFromDate" : "2023/01/01", //ISO format - yyyy-mm-dd
    "effectiveToDate" : "2023/12/31",
    ..
    ..
    "insurancePolicyCoverages" : [
      {
        "coverageName" : "Collision",
        "effectiveFromDate" : "2023/01/01",
        "effectiveToDate" : "2023/12/31",
        ..
        ..
      },
      {
        "coverageName" : "Theft Protection",
        "effectiveFromDate" : "2023/01/01",
        "effectiveToDate" : "2023/12/31",
        ..
        ..
      }
    ]
  },
  "insurancePolicyAssets" : [
    {
      "assetName" : "Audi A3",
      "effectiveFromDate" : "2023/01/01",
      "effectiveToDate" : "2023/12/31",
      ..
      ..
      "insurancePolicyCoverages" : [
        {
          "coverageName" : "Collision",
          "effectiveFromDate" : "2023/01/01",
          "effectiveToDate" : "2023/12/31",
          ..
          ..
        },
        {
          "coverageName" : "Theft Protection",
          "effectiveFromDate" : "2023/01/01",
          "effectiveToDate" : "2023/12/31",
          ..
          ..
        }
      ]
    }
  ]
}
```

```

    ]
  },
  {
    "assetName" : "Audi A4",
    "effectiveFromDate" : "2023/01/01",
    "effectiveToDate" : "2023/12/31",
    ..
    ..
    "insurancePolicyCoverages" : [
      {
        "coverageName" : "Collision",
        "effectiveFromDate" : "2023/01/01",
        "effectiveToDate" : "2023/12/31",
        ..
        ..
      },
      {
        "coverageName" : "Theft Protection",
        "effectiveFromDate" : "2023/01/01",
        "effectiveToDate" : "2023/12/31",
        ..
        ..
      }
    ]
  }
]
}

```

This example shows a sample response when you pass the `contextOnly` query parameter as true in the request. When the `contextOnly` parameter is true, the API returns a context ID along with the insurance policy details.

```

{
  "contextId": "e02036acf464f323f197c200895fec48a6a62ae3c711b3603609f08817c4af48",
  "insurancePolicy": {}
}

```

This example shows a sample response when `expand[attributes]` query parameter is set to true.

```

{
  "insurancePolicy": {
    "effectiveFromDate": "2024-01-01",
    "effectiveToDate": "2024-09-29",
    "originalEffectiveFromDate": "2024-01-01",
    "originalEffectiveToDate": "2024-12-31",
    "active": true,
    ....
    ....
    "grossWrittenPremium": 1678.28,
    "insurancePolicyAssets": [
      {
        "effectiveFromDate": "2024-01-01",
        "effectiveToDate": "2024-09-29",

```

```

"assetName": "Auto Bundle",
"id": "0YWSG0000000N4f4AE",
....
....

"insurancePolicyAssets": [],
"insurancePolicyParticipants": [
  {
    "effectiveFromDate": "2024-01-01",
    "effectiveToDate": "2024-09-29",
    "id": "0aoSG000000sNvhYAE",
    ....
    ....

    "insurancePolicyAssets": [],
    "insurancePolicyCoverages": [],
    "insurancePolicyParticipants": [],
    "insurancePolicyParticipantAttributes": [
      {
        "id": "19USG0000004r382AA",
        "attributeDefinitionId": "0tjSG00000049D8YAI",
        "attributeName": "DriverAccidents",
        "attributeValue": "2.0",
        "insurancePolicyParticipantId": "0aoSG000000sNvhYAE"
      }
    ],
    "additionalFields": {}
  }
],
"insurancePolicyCoverages": [
  {
    "coverageName": "Comprehensive",
    "id": "0cYSG0000005gt32AA",
    "insurancePolicyAssetId": "0YWSG0000000N4f4AE",
    "insurancePolicyId": "0YTSG0000000233R4AQ",
    "name": "IPC-00000034",
    ....
    ....

    "insurancePolicyCoverageAttributes": [
      {
        "id": "19VSG0000004r2w2AA",
        "attributeDefinitionId": "0tjSG00000049DjYAI",
        "attributeName": "LimitBI",
        "attributeValue": "100",
        "insurancePolicyCoverageId": "0cYSG0000005gt32AA"
      },
      ....
      ....

      {
        "id": "19VSG0000004r342AA",
        "attributeDefinitionId": "0tjSG00000049DXYAY",
        "attributeName": "RentalNumDays",
        "attributeValue": "10",
        "insurancePolicyCoverageId": "0cYSG0000005gt32AA"
      }
    ],

```

```

        "additionalFields": {}
      },
      ],
      "insurancePolicyAssetAttributes": [
        {
          "id": "19QSG0000004r2X2AQ",
          "attributeDefinitionId": "0tjSG00000049E4YAI",
          "attributeName": "Body Type",
          "attributeValue": "NA",
          "insurancePolicyAssetId": "0YWSG0000000N4f4AE"
        },
        ....
        ....
        {
          "id": "19QSG0000004r2Y2AQ",
          "attributeDefinitionId": "0tjSG00000049DIYAY",
          "attributeName": "Alternative Fuel",
          "attributeValue": "NA",
          "insurancePolicyAssetId": "0YWSG0000000N4f4AE"
        }
      ],
      "additionalFields": {}
    }
  ],
  "additionalFields": {}
}

```

This example shows a sample response for the get policy request with additional fields. The request returns a list of field-set IDs when the operator is \$in and a single field-set ID when the operator is \$eq.

```

// GET
/med/insurance/policies/MS3000023R4Q?expand[additionalFields][fieldSet][$in]=0XSG00001AD34G,0XSG00001AUF4G,0XSG00001AIP4G,0XSG00001A5H4W

{
  "insurancePolicy": {
    "effectiveFromDate": "2024-01-01",
    "effectiveToDate": "2024-09-29",
    ....
    ....
    "insurancePolicyAssets": [
      {
        "effectiveFromDate": "2024-01-01",
        "effectiveToDate": "2024-09-29",
        "assetName": "Auto Bundle",
        "id": "0YWSG0000000N4f4AE",
        "assetCode": "BMW",
        ....
      },
      "insurancePolicyAssets": [],
      "insurancePolicyParticipants": [
        {
          "effectiveFromDate": "2024-01-01",

```

```

        "effectiveToDate": "2024-09-29",
        "id": "0aoSG000000sNvhYAE",
        "insurancePolicyId": "0YTSG000000233R4AQ",
        "name": "IPP-00000032",
        ....
        ....

        "insurancePolicyParticipantAttributes": [],
        "additionalFields": {
            "InsurancePolicyId": "0YTSG000000233R4AQ",
            "EffectiveFromDate": "2024-01-01",
            "Id": "0aoSG000000sNvhYAE"
        },
    },
],
"insurancePolicyCoverages": [
    {
        "coverageName": "Comprehensive",
        "id": "0cYSG0000005gt32AA",
        "insurancePolicyAssetId": "0YWSG0000000N4f4AE",
        ....

    },
    "insurancePolicyCoverageAttributes": [],
    "additionalFields": {
        "CoverageName": "Comprehensive",
        "Id": "0cYSG0000005gt32AA"
    }
}
],
"insurancePolicyAssetAttributes": [],
"additionalFields": {
    "Name": "IPA-00000035"
}
}
],
"additionalFields": {
    "CheckboxField__c": true,
    "EffectiveFromDate": "2024-01-01",
    "Name": "Admin User Policy001"
}
}
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
contextId	String	ID of the context.	Small, 63.0	63.0
error	Error Response	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
insurancePolicy	Map<String, Object>	Details of insurance policy.	Small, 63.0	63.0

Policy

Output representation of policy details.

JSON example

This request shows a sample success response.

```
{
  "policyId": "0YTx0000000253GAA"
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
originalPolicyId	String	ID of the original policy.	Small, 63.0	63.0
policyId	String	ID of the issued or endorsed policy.	Small, 63.0	63.0

Refund

Output representation of refund details.

JSON example

```
refund": {
  "fee": 15.71,
  "premium": 31.42,
  "tax": 15.71
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
fee	Double	Refund fee amount after cancellation.	Small, 63.0	63.0
premium	Double	Refund premium amount after cancellation.	Small, 63.0	63.0
tax	Double	Refund tax amount after cancellation.	Small, 63.0	63.0

Insurance Policy Standard Invocable Actions

Learn more about the standard invocable actions available with Insurance Policy.

[Cancel Insurance Policy Action](#)

Cancel an insurance policy by using a set of user inputs that represent policy details.

[Endorse Insurance Policy Action](#)

Endorse an insurance policy by using a set of user inputs that represent policy details.

[Get Insurance Policy Action](#)

Get the details of an insurance policy.

[Issue Insurance Policy Action](#)

Issue an insurance policy by using a set of user inputs that represent policy details.

[Renew Insurance Policy Action](#)

Renew an insurance policy by using a set of user inputs that represent policy details.

[Renew Insurance Policies Action](#)

Process the renewal of insurance policies in bulk.

Cancel Insurance Policy Action

Cancel an insurance policy by using a set of user inputs that represent policy details.

This action is available in API version 63.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v64.0/actions/standard/cancelInsurancePolicy`

Formats

JSON

HTTP Methods

POST

Authentication

Authorization: Bearer token

Inputs

Input	Details
<code>cancelPolicyIAInputRep</code>	Type Apex class Description An Apex ConnectApi.CancelPolicyIAInputRep record that contains details of the policy to be canceled.

Outputs

Output	Details
<code>cancelPolicyIAOutputRep</code>	Type Apex class

Output	Details
	Description An Apex ConnectApi.CancelPolicyIAOutputRep record that contains the details of the policy that was canceled.

Usage

JSON Sample Request

```
{
  "inputs": [
    {
      "CancelPolicyIAInputRep": {
        "policyId": "0YTxx00000001H3GAI",
        "effectiveDate": "2024-09-08",
        "transactionRecord": {
          "name": "Cancel Transaction"
        }
      }
    }
  ]
}
```

Endorse Insurance Policy Action

Endorse an insurance policy by using a set of user inputs that represent policy details.
This action is available in API version 63.0 and later.

Supported REST HTTP Methods

URI

/services/data/v64.0/actions/standard/endorseInsurancePolicy

Formats

JSON

HTTP Methods

POST

Authentication

Authorization: Bearer token

Inputs

Input	Details
endorsePolicyIAInputRep	<p>Type</p> <p>Apex class</p> <p>Description</p> <p>An Apex ConnectApi.EndorsePolicyIAInputRep record that contains the details of the policy to be endorsed.</p>

Outputs

Output	Details
endorsedPolicyId	<p>Type</p> <p>String</p> <p>Description</p> <p>The ID of the endorsed insurance policy.</p>

Usage

JSON Sample Request

```
{
  "inputs": [
    {
      "EndorsePolicyIAInputRep": {
        "contextId":
"5dbc40312f69b2d8cb4e7042a700d06f32fc5b9df7de3eabc94aa19a66ed77b3",
        "policyId": "0YTx00000001IfGAI",
        "effectiveDate" : "2024-06-06",
        "insurancePolicy": {
          "policyName": "Endorsed Test Auto Ins policy",
          "policyNumber": "PLC5678987"
        },
        "transactionRecord": {
          "name": "Endorsed Policy Transaction"
        },
        "additionalInput": [
          {
            "instanceKey": "AutoBundle2",
            "additionalFieldsList": [
              {
                "key":"AssetDescription__c",
                "value": "Test_AssetDescription__c"
              },
              {
```

```

        "key": "SourceSystem",
        "value": "Test_SourceSystem"
    }
}
},
{
    "instanceKey": "AutoDriver",
    "additionalFieldsList": [
        {
            "key": "ParticipantName",
            "value": "Test_ParticipantName"
        },
        {
            "key": "Participant_Hometown__c",
            "value": "Test_Participant_Hometown__c"
        }
    ]
},
{
    "instanceKey": "AutoRoot",
    "additionalFieldsList": [
        {
            "key": "Primary_Policy_Holder__c",
            "value": "Test_Primary_Policy_Holder__c"
        },
        {
            "key": "PolicyDescription",
            "value": "Test_PolicyDescription"
        }
    ]
},
{
    "instanceKey": "AutoBundle2_AutoComp1",
    "additionalFieldsList": [
        {
            "key": "Coverage_Description__c",
            "value": "Test_Coverage_Description__c"
        },
        {
            "key": "Description",
            "value": "Test_Description"
        }
    ]
}
}
]
}
]
}
}

```

Get Insurance Policy Action

Get the details of an insurance policy.

This action is available in API version 63.0 and later.

Supported REST HTTP Methods

URI

/services/data/v64.0/actions/standard/getInsurancePolicy

Formats

JSON

HTTP Methods

POST

Authentication

Authorization: Bearer token

Inputs

Input	Details
insurancePolicyId	Type String Description The ID of the insurance policy to be retrieved.

Outputs

Output	Details
contextId	Type Apex class Description The context ID of the retrieved insurance policy.

Usage

JSON Sample Request

```
{
  "inputs": [
    {
      "policyId": "0YTx00000001H3GAI"
    }
  ]
}
```

```
}  
  ]  
}
```

Issue Insurance Policy Action

Issue an insurance policy by using a set of user inputs that represent policy details.

This action is available in API version 63.0 and later.

Supported REST HTTP Methods

URI

`/services/data/v64.0/actions/standard/issueInsurancePolicy`

Formats

JSON

HTTP Methods

POST

Authentication

Authorization: Bearer token

Inputs

Input	Details
<code>issuePolicyIAInputRep</code>	Type Apex class Description An Apex ConnectApi.IssuePolicyIAInputRep record that contains the details of the policy to be issued.

Outputs

Output	Details
<code>issuedPolicyId</code>	Type String Description The ID of the issued policy.

Usage

JSON Sample Request

```
{
  "inputs": [
    {
      "IssuePolicyIAInputRep": {
        "contextId":
"c98a08a0a9725b7a7d42c35d02e49d80c77a5aae4ea687b0bf09a7997d2af3d3",
        "insurancePolicy": {
          "policyName": "Test Auto Ins policy Via REST",
          "policyNumber": "PLC5678987",
          "effectiveFromDate" : "2024-01-01",
          "effectiveToDate" : "2024-12-31",
          "standardFee": "1011"
        },
        "transactionRecord": {
          "name": "Sold Policy Transaction"
        },
        "additionalInput": [
          {
            "instanceKey": "AutoBundle2",
            "additionalFieldsList": [
              {
                "key": "AssetDescription__c",
                "value": "Test_AssetDescription__c"
              },
              {
                "key": "SourceSystem",
                "value": "Test_SourceSystem"
              }
            ]
          },
          {
            "instanceKey": "AutoDriver",
            "additionalFieldsList": [
              {
                "key": "ParticipantName",
                "value": "Test_ParticipantName"
              },
              {
                "key": "Participant_Hometown__c",
                "value": "Test_Participant_Hometown__c"
              }
            ]
          },
          {
            "instanceKey": "AutoRoot",
            "additionalFieldsList": [
              {
                "key": "Primary_Policy_Holder__c",
                "value": "Test_Primary_Policy_Holder__c"
              },
              {

```

```
        "key": "PolicyDescription",
        "value": "Test_PolicyDescription"
      }
    ],
    },
    {
      "instanceKey": "AutoBundle2_AutoComp1",
      "additionalFieldsList": [
        {
          "key": "Coverage_Description__c",
          "value": "Test_Coverage_Description__c"
        },
        {
          "key": "Description",
          "value": "Test_Description"
        }
      ]
    }
  ]
}
}
```

Renew Insurance Policy Action

Renew an insurance policy by using a set of user inputs that represent policy details.
This action is available in API version 63.0 and later.

Supported REST HTTP Methods

URI

/services/data/v64.0/actions/standard/renewInsurancePolicy

Formats

JSON

HTTP Methods

POST

Authentication

Authorization: Bearer token

Inputs

Input	Details
renewPolicyIAInputRep	Type Apex class

Input	Details
	Description An Apex ConnectApi.RenewPolicyIAInputRep record that contains the details of the policy to be renewed.

Outputs

Output	Details
renewedPolicyId	Type String Description The ID of the renewed policy.

Usage

JSON Sample Request

```
{
  "inputs": [
    {
      "RenewPolicyIAInputRep": {
        "contextId": "d767b27ec095f9f95c5a34299e3bd6f507df23e8d0692dfed17f98dcb5b661b6",
        "policyId": "0YTxx00000001ovGAA",
        "insurancePolicy": {
          "policyName": "Renewed_Auto_Policy_00002",
          "policyNumber": "Renewed Auto Policy 00002",
          "effectiveFromDate": "2025-01-01",
          "effectiveToDate": "2025-12-31"
        },
      },
      "transactionRecord": {
        "name": "Renew Transaction"
      },
      "additionalInput": [
        {
          "instanceKey": "AutoBundle2",
          "additionalFieldsList": [
            {
              "key": "AssetDescription__c",
              "value": "Test_AssetDescription__c"
            },
            {
              "key": "SourceSystem",
              "value": "Test_SourceSystem"
            }
          ]
        }
      ]
    },
    {

```

```

    "instanceKey": "AutoDriver",
    "additionalFieldsList": [
      {
        "key": "ParticipantName",
        "value": "Test_ParticipantName"
      },
      {
        "key": "Participant_Hometown__c",
        "value": "Test_Participant_Hometown__c"
      }
    ]
  },
  {
    "instanceKey": "AutoRoot",
    "additionalFieldsList": [
      {
        "key": "Primary_Policy_Holder__c",
        "value": "Test_Primary_Policy_Holder__c"
      },
      {
        "key": "PolicyDescription",
        "value": "Test_PolicyDescription"
      }
    ]
  },
  {
    "instanceKey": "AutoBundle2_AutoComp1",
    "additionalFieldsList": [
      {
        "key": "Coverage_Description__c",
        "value": "Test_Coverage_Description__c"
      },
      {
        "key": "Description",
        "value": "Test_Description"
      }
    ]
  }
]
}
]
}
]
}
}

```

Renew Insurance Policies Action

Process the renewal of insurance policies in bulk.

This action is available in API version 64.0 and later.

Supported REST HTTP Methods

URI

/services/data/v64.0/actions/standard/renewInsurancePolicies

Formats

JSON, XML

HTTP Methods

POST

Authentication

Authorization: Bearer **token**

Inputs

Input	Details
renewInsurancePoliciesIAInputRep	<p>Type</p> <p>Apex-defined</p> <p>Description</p> <p>Required. A collection of Apex input records that contain details of the insurance policies to be renewed. See ConnectApi.RenewInsurancePoliciesIAInputRep on page 114 for the list of input parameters.</p>

Outputs

No output representation as it's an async invocable action, which publishes the request to a queue for processing.

Example

POST

This example shows a sample request for the Renew Insurance Policies action.

```
{
  "inputs": [
    {
      "RenewInsurancePoliciesIAInputRep": {
        "policyId": "0YTIU000000xUi14AE",
        "jobIdentifier": "testJob368",
        "executeConfigurationRules": true,
        "executeQualificationRules": true,
        "issueQuote": false,
        "sameTenure": true
        "priceBookId": "01sxx0000005pyfAAA",
        "productSellingModel": "One Time",
        "flowAPINameWithNamespace": "abc",
        "fieldSetIds" : {
          "policyFieldSetId": "0YTIU000000xUi14AE",
          "assetFieldSetId" : "0YTIU000000xUi14AE",

```

```

        "participantFieldSetId" : "0YTIU000000xUi14AE",
        "coverageFieldSetId" : "0YTIU000000xUi14AE",
    }
}
}
]
}

```

Insurance Policy Administration Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

[Flow for Insurance Policy Administration](#)

Represents the metadata associated with a flow. Use Flow to create an application that takes users through a series of pages to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

Flow for Insurance Policy Administration

Represents the metadata associated with a flow. Use Flow to create an application that takes users through a series of pages to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Insurance Policy Administration exposes additional `actionType` values for the FlowActionCall Metadata type.

Field Name	Field Type	Description
<code>actionType</code>	InvokableActionType (enumeration of type string)	<p>Required.</p> <p>The action type. Additional valid values only for Insurance Policy Administration include:</p> <ul style="list-style-type: none"> <code>cancelInsurancePolicy</code>—Cancel an insurance policy by using a set of user inputs that represent policy details. <code>endorseInsurancePolicy</code>—Endorse an insurance policy by using a set of user inputs that represent policy details. <code>getInsurancePolicy</code>—Get the details of an insurance policy. <code>issueInsurancePolicy</code>—Issue an insurance policy by using a set of user inputs that represent policy details. <code>renewInsurancePolicy</code>—Renew an insurance policy by using a set of user inputs that represent policy details.

Insurance Policy Administration Apex Reference

Apex is a strongly typed, object-oriented programming language that allows developers to execute flow and transaction control statements on the Salesforce Platform server, in conjunction with calls to the API. Insurance Policy Administration Apex reference includes built-in Apex classes and interfaces, grouped by namespace.

[ConnectApi Namespace](#)

The `ConnectApi` namespace (also called Connect in Apex) in Insurance Policy Administration provides classes for accessing the same data available in the Insurance Policy Administration Business APIs.

ConnectApi Namespace

The `ConnectApi` namespace (also called Connect in Apex) in Insurance Policy Administration provides classes for accessing the same data available in the Insurance Policy Administration Business APIs.

For information about working with the `ConnectApi` classes, see [Connect in Apex](#).

[InsurancePolicyAdminFamily Class](#)

Manage insurance policies comprehensively, for different transaction types, such as issuance, endorsement, renewal, and cancellation.

[ConnectApi Input Classes](#)

Some `ConnectApi` methods take arguments that are instances of `ConnectApi` input classes.

[ConnectApi Output Classes](#)

Most `ConnectApi` methods return instances of `ConnectApi` output classes.

InsurancePolicyAdminFamily Class

Manage insurance policies comprehensively, for different transaction types, such as issuance, endorsement, renewal, and cancellation.

Namespace

`ConnectApi`

[InsurancePolicyAdminFamily Methods](#)

These methods are for `InsurancePolicyAdminFamily`. All methods are static.

InsurancePolicyAdminFamily Methods

These methods are for `InsurancePolicyAdminFamily`. All methods are static.

[CancelPolicy\(CancelPolicyInput, policyId\)](#)

Cancel an insurance policy version. The API takes the effective date of cancellation from the input JSON. You can advance the cancellation date within the policy term, but you can't postpone the cancellation.

[EndorsePolicy\(EndorsePolicyInput, policyId\)](#)

Endorse the latest version of the insurance policy based on the context ID and effective date. You can also include other optional fields and additional details in the input JSON.

[GetPolicy\(policyId, connectDynamicParameters\)](#)

Get details of an insurance policy. Pass additional context or metadata to customize the information that's fetched.

[IssuePolicy\(IssuePolicyInput\)](#)

Use the context ID provided in the input JSON to generate a new insurance policy.

[RenewPolicy\(RenewPolicyInput, policyId\)](#)

Use the context ID of an existing policy to renew a policy. Modify the renewal date, and add optional fields and additional information in the input JSON.

CancelPolicy(CancelPolicyInput, policyId)

Cancel an insurance policy version. The API takes the effective date of cancellation from the input JSON. You can advance the cancellation date within the policy term, but you can't postpone the cancellation.

API Version

63.0

Requires Chatter

No

Signature

```
public static ConnectApi.CancelPolicyOutputRepresentation  
CancelPolicy(ConnectApi.CancelPolicyInputRepresentation CancelPolicyInput, String  
policyId)
```

```
ConnectApi.InsurancePolicyAdminFamily, CancelPolicy,  
[ConnectApi.CancelPolicyInputRepresentation, String],  
ConnectApi.CancelPolicyOutputRepresentation
```

Parameters

CancelPolicyInput

Type: [ConnectApi.CancelPolicyInputRepresentation](#) on page 110

Description

policyId

Type: String

Input representation of the request to cancel a policy.

Return Value

Type: [ConnectApi.CancelPolicyOutputRepresentation](#) on page 115

Usage

Send the cancellation effective date and transaction record in the request to cancel a policy.

EndorsePolicy(EndorsePolicyInput, policyId)

Endorse the latest version of the insurance policy based on the context ID and effective date. You can also include other optional fields and additional details in the input JSON.

API Version

63.0

Requires Chatter

No

Signature

```
public static ConnectApi.PolicyOutputRepresentation  
EndorsePolicy(ConnectApi.EndorsePolicyInputRepresentation EndorsePolicyInput, String  
policyId)
```

```
ConnectApi.InsurancePolicyAdminFamily, EndorsePolicy,  
[ConnectApi.EndorsePolicyInputRepresentation, String], ConnectApi.PolicyOutputRepresentation
```

Parameters

EndorsePolicyInput

Type: [ConnectApi.EndorsePolicyInputRepresentation](#) on page 111

Input representation of the request to endorse a policy.

policyId

Type: String

ID of the policy to be endorsed.

Return Value

Type: [ConnectApi.PolicyOutputRepresentation](#) on page 116

GetPolicy(policyId, connectDynamicParameters)

Get details of an insurance policy. Pass additional context or metadata to customize the information that's fetched.

API Version

63.0

Requires Chatter

No

Signature

```
public static ConnectApi.GetPolicyOutputRepresentation GetPolicy(String policyId,
Map<String,String> connectDynamicParameters)
```

```
ConnectApi.InsurancePolicyAdminFamily, GetPolicy, [String, Map<String,String>],
ConnectApi.GetPolicyOutputRepresentation
```

Parameters

policyId

Type: String

ID of the policy for which the details are to be fetched.

connectDynamicParameters

Type: Map<String,String>

Map of connect dynamic parameters to fetch the policy details.

Return Value

Type: [ConnectApi.GetPolicyOutputRepresentation](#) on page 116

Usage

Use this API to fetch the insurance policy details. This includes related assets, participants, and coverages. Use query parameters to request additional information.

- When the `contextOnly` parameter is true, the API returns a context ID along with the insurance policy details.
- When you send a request with additional fields with the operator `$in`, it returns a list of field-set IDs. When the operator is `$eq`, the request returns a single field-set ID.

For examples, see [Get Policy](#).

IssuePolicy(IssuePolicyInput)

Use the context ID provided in the input JSON to generate a new insurance policy.

API Version

63.0

Requires Chatter

No

Signature

```
public static ConnectApi.PolicyOutputRepresentation
IssuePolicy(ConnectApi.IssuePolicyInputRepresentation IssuePolicyInput)
```

```
ConnectApi.InsurancePolicyAdminFamily, IssuePolicy,
[ConnectApi.IssuePolicyInputRepresentation], ConnectApi.PolicyOutputRepresentation
```


Parameters

*IssuePolicyInput*Type: [ConnectApi.IssuePolicyInputRepresentation](#)

Input representation of the request to issue an insurance policy.

Return Value

Type: [ConnectApi.PolicyOutputRepresentation](#) on page 116**RenewPolicy(RenewPolicyInput, policyId)**

Use the context ID of an existing policy to renew a policy. Modify the renewal date, and add optional fields and additional information in the input JSON.

API Version

63.0

Requires Chatter

No

Signature

```
public static ConnectApi.PolicyOutputRepresentation
RenewPolicy (ConnectApi.RenewPolicyInputRepresentation RenewPolicyInput, String policyId)
```

```
ConnectApi.InsurancePolicyAdminFamily, RenewPolicy,
[ConnectApi.RenewPolicyInputRepresentation, String], ConnectApi.PolicyOutputRepresentation
```

Parameters

*RenewPolicyInput*Type: [ConnectApi.RenewPolicyInputRepresentation](#) on page 113

Input representation of the request to renew a policy.

policyId

Type: String

ID of the insurance policy that's to be renewed.

Return Value

Type: [ConnectApi.PolicyOutputRepresentation](#) on page 116

ConnectApi Input Classes

Some `ConnectApi` methods take arguments that are instances of `ConnectApi` input classes.

Input classes are concrete unless marked abstract in this documentation. Concrete input classes have public constructors that have no parameters.

Some methods have parameters that are typed with an abstract class. You must pass in an instance of a concrete child class for these parameters.

Most input class properties can be set. Read-only properties are noted in this documentation.

Insurance Policy Administration includes these Apex input classes.

[ConnectApi.AdditionalFieldsMapValue](#)

AdditionalFieldsMapValue wrapper object for use as child wrapper of AdditionalFieldsMap.

[ConnectApi.CancelPolicyInputRepresentation](#)

Input representation of the request to cancel a policy.

[ConnectApi.EndorsePolicyInputRepresentation](#)

Input representation of the request to endorse a policy. Context ID of the quote that's used for endorsement and date that the endorsed policy version is effective from must be passed in the request.

[ConnectApi.InsurancePolicyInputRepresentation](#)

Input representation details of an insurance policy.

[ConnectApi.InsurancePolicyTransactionInputRepresentation](#)

Input representation details of an insurance policy transaction.

[ConnectApi.IssuePolicyInputRepresentation](#)

Input representation of the request to issue an insurance policy.

[ConnectApi.RenewPolicyInputRepresentation](#)

Input representation of the request to renew a policy.

[ConnectApi.FieldSetIDsInputRep](#)

List of the field set IDs for asset, coverage, participant, and policy objects.

[ConnectApi.RenewInsurancePoliciesInputRep](#)

An Apex ConnectApi.RenewInsurancePoliciesInputRep record that contains the details of the insurance policies to be renewed.

ConnectApi.AdditionalFieldsMapValue

AdditionalFieldsMapValue wrapper object for use as child wrapper of AdditionalFieldsMap.

Property	Type	Description	Required or Optional	Available Version
additionalFieldsMapValue	Object	Additional fields map containing fields to be updated.	Required	63.0

ConnectApi.CancelPolicyInputRepresentation

Input representation of the request to cancel a policy.

Property	Type	Description	Required or Optional	Available Version
effectiveDate	String	Date on which the cancellation of the policy goes into effect.	Required	63.0

Property	Type	Description	Required or Optional	Available Version
transactionRecord	InsurancePolicyTransaction on page 112	The JSON structure that overrides the transaction information.	Optional	63.0

ConnectApi.EndorsePolicyInputRepresentation

Input representation of the request to endorse a policy. Context ID of the quote that's used for endorsement and date that the endorsed policy version is effective from must be passed in the request.

Property	Type	Description	Required or Optional	Available Version
additionalFields	ContextAdditionalFields on page 110>	The JSON structure that's used to add custom or standard fields to the <code>InsurancePolicy</code> , <code>InsurancePolicyAsset</code> , <code>InsurancePolicyCoverage</code> , and <code>InsurancePolicyParticipant</code> objects. The values for <code>additionalFields</code> take precedence over the context or the <code>insurancePolicy</code> JSON. This is applicable only for the endorsed policy version. Address and Geolocation field types are not supported as additional fields.	Optional	63.0
contextId	String	Context ID of the quote that's used for endorsement.	Required	63.0
effectiveDate	String	Date from when endorsed policy version is effective from.	Required	63.0
insurancePolicy	InsurancePolicyRepresentation on page 112	The JSON structure that overrides the name and policy number of the endorsed policy version.	Optional	63.0
transactionRecord	InsurancePolicyTransaction on page 112	The JSON structure that overrides the transaction information.	Optional	63.0

ConnectApi.InsurancePolicyInputRepresentation

Input representation details of an insurance policy.

Property	Type	Description	Required or Optional	Available Version
effectiveFromDate	String	Date from when the policy is effective.	Optional	63.0
effectiveToDate	String	Date until when the policy is effective.	Optional	63.0

Property	Type	Description	Required or Optional	Available Version
policyName	String	Name of the insurance policy.	Optional	63.0
policyNumber	String	Number of the policy.	Optional	63.0

ConnectApi.InsurancePolicyTransactionInputRepresentation

Input representation details of an insurance policy transaction.

Property	Type	Description	Required or Optional	Available Version
name	String	Name of the transaction, such as Cancel Transaction.	Optional	63.0

ConnectApi.IssuePolicyInputRepresentation

Input representation of the request to issue an insurance policy.

Property	Type	Description	Required or Optional	Available Version
additionalFields	ContextIdField on page 110	The JSON structure that's used to add custom or standard fields to the InsurancePolicy, InsurancePolicyAsset, InsurancePolicyCoverage, and InsurancePolicyParticipant objects. The values for additionalFields take precedence over the context or the insurancePolicy JSON. Address and Geolocation field types are not supported as additional fields. Add the SourceQuoteld standard field to the additionalFields option when the context ID in the input JSON for the Issue Insurance Policy API is generated by the Create Insurance Quote API.	Optional	63.0
contextId	String	Context ID of the quote that's used to create the policy.	Required	63.0
insurancePolicy	InsurancePolicyInputRepresentation on page 111	The JSON structure that's used to create the policy. It contains fields and values from the Insurance Policy object.	Optional	63.0
transactionRecord	InsurancePolicyTransactionInputRepresentation on page 112	The JSON structure that's used to override transaction information.	Optional	63.0

ConnectApi.RenewPolicyInputRepresentation

Input representation of the request to renew a policy.

Property	Type	Description	Required or Optional	Available Version
additionalFields	ConnectApi.AdditionalFields on page 110>	The JSON structure that's used to add custom or standard fields to the <code>InsurancePolicy</code> , <code>InsurancePolicyAsset</code> , <code>InsurancePolicyCoverage</code> , and <code>InsurancePolicyParticipant</code> objects. The values for <code>additionalFields</code> take precedence over the context or the <code>insurancePolicy</code> JSON. This is applicable only for the endorsed policy version. Address and Geolocation field types are not supported as additional fields.	Optional	63.0
contextId	String	Context ID of the quote that's used to renew the policy.	Required	63.0
insurancePolicy	InsurancePolicyInputRepresentation on page 111	The JSON structure that's used to renew the policy. It contains fields and values from the Insurance Policy object.	Optional	63.0
transactionRecord	InsurancePolicyInputRepresentation on page 112	The JSON structure that's used to override the transaction information.	Optional	63.0

ConnectApi.FieldSetIdsInputRep

List of the field set IDs for asset, coverage, participant, and policy objects.

Property	Type	Description	Required or Optional	Available Version
asset FieldSetId	String	ID of the field set to get the fields associated with the asset object.	Optional	64.0
coverage FieldSetId	String	ID of the field set to get the fields associated with the coverage object.	Optional	64.0
participant FieldSetId	String	ID of the field set to get the fields associated with the participant object.	Optional	64.0
policy FieldSetId	String	ID of the field set to get the fields associated with the policy object.	Optional	64.0

ConnectApi.RenewInsurancePoliciesInputRep

An Apex ConnectApi.RenewInsurancePoliciesInputRep record that contains the details of the insurance policies to be renewed.

Property	Type	Description	Required or Optional	Available Version
<code>executeConfigurationRules</code>	Boolean	Indicates whether the configuration rules for renewal must be executed (<code>true</code>) or not (<code>false</code>).	Optional	64.0
<code>executeQualificationRules</code>	Boolean	Indicates whether the qualification rules for renewal must be executed (<code>true</code>) or not (<code>false</code>).	Required for quote renewals.	64.0
<code>fieldSetIds</code>	Field SetIdsInputRep on page 113	Input representation of field set IDs.	Optional	64.0
<code>flowAPINameWithNamespace</code>	String	The API name of the flow to execute changes before each policy or quote renewal process. Include the namespace prefix, if any.	Optional	64.0
<code>issueQuote</code>	Boolean	Indicates whether the renewal quote must be issued (<code>true</code>) or not (<code>false</code>).	Optional	64.0
<code>jobIdentifier</code>	String	A unique ID of the batch job that's used to monitor and generate reports using Custom Report Type. The maximum length is 50 characters.	Required	64.0
<code>policyId</code>	String	ID of the insurance policy to be renewed.	Required	64.0
<code>priceBookId</code>	String	ID of the price book that's used to renew policies. Defaults to the Standard Price Book record ID if no value is provided.	Required for quote renewals.	64.0
<code>productSellingModel</code>	String	The model that's used to offer the product, such as one-time or term-based. The default is one-time if no value is provided.	Required for quote renewals.	64.0
<code>sameTenure</code>	Boolean	Indicates whether the same tenure must be set for renewal as original policy (<code>true</code>) or not (<code>false</code>).	Optional	64.0

ConnectApi Output Classes

Most ConnectApi methods return instances of ConnectApi output classes.

All properties are read-only, except for instances of output classes created within test code.

All output classes are concrete unless marked abstract in this documentation.

All concrete output classes have no-argument constructors that you can invoke only from test code. See [Testing ConnectApi Code](#).

Insurance Policy Administration includes these Apex output classes.

[ConnectApi.CancelPolicyOutputRepresentation](#)

Output representation details of cancel policy, such as impacted policy versions and refund associated with the cancellation.

[ConnectApi.ErrorDetailsOutputRepresentation](#)

Output representation of the error details, such as error code and error message.

[ConnectApi.ErrorOutputRepresentation](#)

Output representation details of the error encountered while processing the request.

[ConnectApi.GetPolicyOutputRepresentation](#)

Output representation of the policy details.

[ConnectApi.PolicyOutputRepresentation](#)

Output representation of the policy details.

[ConnectApi.RefundOutputRepresentation](#)

Output representation of the refund payment.

ConnectApi.CancelPolicyOutputRepresentation

Output representation details of cancel policy, such as impacted policy versions and refund associated with the cancellation.

Property Name	Type	Description	Available Version
error	ConnectApi.ErrorOutputRepresentation on page 115	List of errors encountered during the processing of the API request.	63.0
impactedPolicyVersions	List<String>	List of impacted policy versions.	63.0
refund	ConnectApi.RefundOutputRepresentation on page 116	Refund associated with cancellation operation.	63.0

ConnectApi.ErrorDetailsOutputRepresentation

Output representation of the error details, such as error code and error message.

Property Name	Type	Description	Available Version
code	String	The error code thrown when the request fails.	63.0
message	String	Message stating the reason for error, if any.	63.0

ConnectApi.ErrorOutputRepresentation

Output representation details of the error encountered while processing the request.

Property Name	Type	Description	Available Version
detail	String	Description of the error.	63.0

Property Name	Type	Description	Available Version
errors	ConnectApi.ErrorDetailsOutputRepresentation on page 115>	Output representation of error details.	63.0
instance	String	The instance where the error occurred.	63.0
title	String	Short label of the error type.	63.0

ConnectApi.GetPolicyOutputRepresentation

Output representation of the policy details.

Property Name	Type	Description	Available Version
contextId	String	ID of the context.	63.0
error	ConnectApi.ErrorDetailsOutputRepresentation on page 115	List of errors encountered during the processing of the API request.	63.0
insurancePolicy	Map<String, Object>	Details of insurance policy.	63.0

ConnectApi.PolicyOutputRepresentation

Output representation of the policy details.

Property Name	Type	Description	Available Version
error	ConnectApi.ErrorDetailsOutputRepresentation on page 115	List of errors encountered during the processing of the API request.	63.0
originalPolicyId	String	ID of the original policy.	63.0
policyId	String	ID of the issued or endorsed policy.	63.0

ConnectApi.RefundOutputRepresentation

Output representation of the refund payment.

Property Name	Type	Description	Available Version
fee	Double	Refund fee amount after cancellation.	63.0
premium	Double	Refund premium amount after cancellation.	63.0
tax	Double	Refund tax amount after cancellation.	63.0

Insurance Policy Developer Journey

To manage insurance policies by using Salesforce Quoting, use the Quoting APIs to generate a quote context, which serves as the foundation for policy operations. For managing policies without Salesforce Quoting, use the Rating API to create a price-adjusted context. This price-adjusted context provides the necessary rated data to manage insurance policies without depending on a formal quote record.

[Create and Manage Insurance Policies by Using Salesforce Quoting](#)

Learn how to effectively use the Insurance Policy and Quoting Connect APIs, along with the associated invocable actions, to build comprehensive user journeys by using Salesforce Flows and Omniscripts.

[Create and Manage Insurance Policies Without Using Salesforce Quoting](#)

Learn how to use the Insurance Policy APIs and invocable actions to manage the policy lifecycle when Salesforce quoting isn't used, or when a third-party system handles the quoting process.

[Cancel Policy](#)

The policy cancellation process addresses the termination of an insurance policy before its scheduled expiration date. Common reasons for canceling a policy include not paying the premiums or a direct cancellation request from the policyholder.

Create and Manage Insurance Policies by Using Salesforce Quoting

Learn how to effectively use the Insurance Policy and Quoting Connect APIs, along with the associated invocable actions, to build comprehensive user journeys by using Salesforce Flows and Omniscripts.

The Salesforce Insurance Policy and Quoting APIs, and the related invocable actions provide a complete set of tools to manage the entire insurance policy lifecycle. These APIs support key policy administration processes, such as policy issuance, endorsement, renewal, and cancellation. By using these APIs and invocable actions, you can create user journeys for insurance representatives who use the Salesforce console, as well as the policyholders and partners who access the services through community portals.

[Endorse Policy](#)

Policy endorsement, also known as a Mid-Term Adjustment (MTA), is the process of making modifications to an active insurance policy during its term. Common modifications include adding or removing coverage, or updating policy attributes, such as limits and deductibles. Endorsements usually affect the policy price and create a new policy version instead of updating the existing record. This new version includes the applied changes, along with a recalculated standard premium. It also contains a prorated term premium that's calculated based on the effective date of the endorsement.

[Issue Policy](#)

The policy issuance process begins after the policyholder accepts the quote provided by the insurance company.

[Renew Policy](#)

Policy renewal is the process of extending insurance coverage for an additional policy term, typically as the current policy approaches its effective-to date. A key part of renewal is that a renewed policy creates a new policy record.

Endorse Policy

Policy endorsement, also known as a Mid-Term Adjustment (MTA), is the process of making modifications to an active insurance policy during its term. Common modifications include adding or removing coverage, or updating policy attributes, such as limits and deductibles. Endorsements usually affect the policy price and create a new policy version instead of updating the existing record. This new version includes the applied changes, along with a recalculated standard premium. It also contains a prorated term premium that's calculated based on the effective date of the endorsement.

Make sure you have these details.

- Policy ID of the active insurance policy to be endorsed.
- Pricebook ID is a required input for generating an endorsement quote.
- Details of the modifications to be made to the policy, such as specific coverages to add or remove, and new limit amounts.

This user journey outlines the process of endorsing a policy, where any changes are initially processed through a quote to assess the pricing impact.

1. Get Policy Context—Use the policy ID to get the context either by:

- Making a GET request on the [Insurance Policy Details](#) Connect API, or
- Using the [Get Insurance Policy](#) invocable action.

2. Create Endorsement Quote—Use the policy context to create the endorsement quote either by:

- Making a POST request on the [Insurance Quote](#) Connect API, or
- Using the [Create Insurance Quote](#) invocable action.

Specify the `pricebookId` during the API call. You can update the header-level fields of the quote, such as the quote name or effective date for the endorsement, by specifying these values as the input parameters during the API call.

A new quote record is generated for the endorsement. After the policyholder confirms the updated quote price, proceed with the next steps in the endorsement workflow.

3. (Optional)Modify Endorsement Quote—If policyholders want to modify the endorsement quote, make a PATCH request on the [Insurance Quote Details](#) Connect API with the required changes.

An updated, price-adjusted context is returned in the response.

4. Endorse Policy—Use the retrieved context to endorse a policy either by:

- Making a POST request on the [Endorse Insurance Policy](#) Connect API, or
- Using the [Endorse Insurance Policy](#) invocable action.

You can change the context fields, such as the policy name or policy number, directly during the API call. You can also provide a name for the transaction record that's created along with the endorsement.

This transaction record documents the financial details and audit information associated with the MTA.

A new endorsed policy version is created.

Expected Results:

- A new policy version record is created in Salesforce, which includes the endorsed changes.
- A new policy version is created with an updated effective-from date. The effective-to date of the existing policy version is adjusted to an end date just before the new version begins, ensuring continuous but non-overlapping coverage.
- The new policy version has the updated premium information, including the new standard premium and the prorated term premium, accurately calculated and stored.
- A transaction record is created and linked to the endorsement activity.

Issue Policy

The policy issuance process begins after the policyholder accepts the quote provided by the insurance company.

1. Create Quote—Create a quote either by:

- Making a POST request on the [Insurance Quote](#) Connect API, or
- Using the [Create Insurance Quote](#) invocable action, or

- Using the [Salesforce Quote User Interface](#).

This process creates the quote and generates the quote ID.

2. Get Quote Context—Use the quote ID to retrieve the quote details either by:

- Making a GET request on the [Insurance Quote Details](#) Connect API, or
- Using the [Get Insurance Quote Details](#) invocable action.

This request returns the context ID in the response that contains the context of the quote.

3. Issue Policy—Use the context details, specifically the context ID, to issue policy either by:

- Making a POST request on the [Issue Insurance Policy](#) Connect API, or
- Using the [Issue Insurance Policy](#) invocable action.

You can provide or override the context data, such as policy name, policy number, effective-from date, effective-to date, and any other additional fields to provide the values for the custom fields during the API call. You can also specify a name for the transaction record that's created as part of the policy issuance process. This transaction record is used to track an audit or as a financial marker.

This request creates a new policy record.

Expected Results:

- A new policy record is created and saved in Salesforce. The policy record includes the details, such as the policy name, number (if provided or created during the API call), policy start date, chosen coverage, and the price.
- Any custom field values that are provided during the API call are saved to the new policy record.
- A transaction record is created and linked to the policy.

Renew Policy

Policy renewal is the process of extending insurance coverage for an additional policy term, typically as the current policy approaches its effective-to date. A key part of renewal is that a renewed policy creates a new policy record.

Make sure you have these details.

- Policy ID of the active insurance policy that's to be renewed.
- Pricebook ID as it's a required input for generating an endorsement quote.

If the policyholder changes their information, it affects the terms or prices of the renewed policy. This journey describes renewing a policy where a formal renewal quote is generated. After the customer confirms the renewal quote, a new policy record is created for the new renewal term.

1. Get Policy Context—Use the policy ID to get the context either by:

- Making a GET request on the [Insurance Policy Details](#) Connect API, or
- Using the [Get Insurance Policy](#) invocable action.

2. Create Renewal Quote—Use the policy context to create a renewal quote either by:

- Making a POST request on the [Insurance Quote](#) Connect API, or
- Using the [Create Insurance Quote Action](#) invocable action.

Specify the `pricebookId` during the API call. You can change the header-level fields of the quote, such as the quote name or the start date of the renewal term by specifying these values as the input parameters during the API call.

A new quote record is generated for the policy renewal. After the customer confirms the updated quote price, proceed with the next steps in the renewal workflow.

3. (Optional)Modify Renewal Quote—If policyholders want to modify the endorsement quote, make a PATCH request on the [Insurance Quote Details](#) Connect API with the desired changes.

An updated, price-adjusted context is returned in the response.

4. Renew Policy—Use the retrieved context to create a new policy record for the renewal term either by:

- Making a POST request on the [Renew Insurance Policy](#) Connect API, or
- Using the [Renew Insurance Policy Action](#) invocable action.

You can change or confirm the policy name, policy number, and effective start and end dates for the renewed policy directly during the API call. Additionally, you can also specify a name for the transaction record that's created along with the renewal record. This transaction record captures the financial details or the audit information related to the renewal record.

Expected Results:

- A brand new policy record is created in Salesforce, representing the coverage for the new policy term. The new policy record includes the details, such as any changed terms, coverages, prices, and the new start and end dates.
- A transaction record is created and linked to the new policy.

Create and Manage Insurance Policies Without Using Salesforce Quoting

Learn how to use the Insurance Policy APIs and invocable actions to manage the policy lifecycle when Salesforce quoting isn't used, or when a third-party system handles the quoting process.

[Endorse Policy](#)

Policy endorsement, also known as a Mid-Term Adjustment (MTA), is a process where you change an active insurance policy. The endorsement process changes the policy version and the premiums. This user journey shows you how to endorse a policy when Salesforce's quoting capabilities aren't used.

[Issue Policy](#)

Issue a new insurance policy when quoting is managed outside Salesforce, by a third-party system. This process results in the creation of a policy record in Salesforce, formally establishing coverage for the customer.

[Renew Policy](#)

Policy renewal extends insurance coverage for an additional term, creating a new policy record. This journey details renewing a policy without using Salesforce quoting, relying on the Rating API for pricing the new term.

Endorse Policy

Policy endorsement, also known as a Mid-Term Adjustment (MTA), is a process where you change an active insurance policy. The endorsement process changes the policy version and the premiums. This user journey shows you how to endorse a policy when Salesforce's quoting capabilities aren't used.

Make sure you have these details.

- Policy ID of the active insurance policy to be endorsed.
- Details of the modifications to be made, such as changes to coverages, limits, and deductibles.

Use the Rating API to generate a price-adjusted context. This price-adjusted context contains the calculated premium and other relevant details that are required to endorse the policy. This process bypasses the need for a formal Salesforce Quote record for pricing.

1. Get Policy Context—Use the policy ID to get the policy context either by:

- Making a GET request on the [Insurance Policy Details](#) Connect API, or
- Using the [Get Insurance Policy](#) invocable action.

2. Generate Price-adjusted Context—Make a POST request on the [Insurance Product Ratings](#) Connect API.

The price-adjusted context record contains the premium impact of the proposed modifications.

3. **(Optional)Run Qualification Rules**—If customers want to determine the policyholder's eligibility for a product, run the qualification rules either by:

- Making a POST request on the [Insurance Quote](#) Connect API, or
- Using the [Create Insurance Quote](#) invocable action.

Set the `saveQuote` parameter to `false`. It doesn't save the quote and returns an updated context with the eligibility details.

4. **Endorse Policy**—Use the retrieved context to endorse a policy either by:

- Making a POST request on the [Endorse Insurance Policy](#) Connect API, or
- Using the [Endorse Insurance Policy](#) invocable action.

You can change the context fields, such as the policy name or policy number, directly during the API call. You can also provide a name for the transaction record that's created along with the endorsement. This transaction record captures the financial details or the audit information related to the MTA.

A new endorsed policy version is created.

Expected Results:

- A new policy version record is created in Salesforce, which includes the endorsed changes.
- A new policy version is created with an updated effective-from date. The effective-to date of the existing policy version is adjusted to an end date just before the new version begins, ensuring continuous but non-overlapping coverage.
- The new policy version has the updated premium information, including the new standard premium and the prorated term premium, accurately calculated and stored.
- A transaction record is created and linked to the endorsement activity.

Issue Policy

Issue a new insurance policy when quoting is managed outside Salesforce, by a third-party system. This process results in the creation of a policy record in Salesforce, formally establishing coverage for the customer.

Use the Rating API to generate a price-adjusted context. This price-adjusted context contains the calculated premium and other relevant details that are required to issue a policy. This process bypasses the need for a formal Salesforce Quote record for pricing.

1. **Generate Price-adjusted Context**—Make a POST request on the [Insurance Product Ratings](#) Connect API.
2. **(Optional)Run Qualification Rules**—If customers want to determine the policyholder's eligibility for a product, run the qualification rules either by:

- Making a POST request on the [Insurance Quote](#) Connect API, or
- Using the [Create Insurance Quote](#) invocable action.

Set the `saveQuote` parameter to `false`. It doesn't save the quote and returns an updated context with the eligibility details.

3. **Issue Policy**—Use the price-adjusted context record to issue a policy either by:

- Making a POST request on the [Issue Insurance Policy](#) Connect API, or
- Using the [Issue Insurance Policy](#) invocable action.

You can provide or override the context data, such as policy name, policy number, effective-from date, effective-to date, and any other additional fields to provide the values for the custom fields during the API call. You can also specify a name for the transaction record that's created as part of the policy issuance process. This transaction record is used to track an audit or as a financial marker.

This request creates a new policy record.

Expected Results:

- A new policy record is created and saved in Salesforce. The policy record includes the details, such as the policy name, number (if provided or created during the API call), policy start date, the coverage you chose, and the price.
- Any custom field values that are provided during the API call are saved to the new policy record.
- A transaction record is created and linked to the policy.

Renew Policy

Policy renewal extends insurance coverage for an additional term, creating a new policy record. This journey details renewing a policy without using Salesforce quoting, relying on the Rating API for pricing the new term.

Make sure you have a valid policy ID of the active insurance policy that's due for renewal.

Use the Rating API to generate a price-adjusted context. This price-adjusted context contains the calculated premium and other relevant details that are required to renew the policy. This process bypasses the need for a formal Salesforce Quote record for pricing.

1. Get Policy Context—Use the policy ID to get the policy context either by:

- Making a GET request on the [Insurance Policy Details](#) Connect API, or
- Using the [Get Insurance Policy](#) invocable action.

2. Generate Price-adjusted Context—Make a POST request on the [Insurance Product Ratings](#) Connect API.

The price-adjusted context record contains the premium impact of the proposed modifications.

3. (Optional)Run Qualification Rules—If customers want to determine the policyholder's eligibility for a product, run the qualification rules either by:

- Making a POST request on the [Insurance Quote](#) Connect API, or
- Using the [Create Insurance Quote](#) invocable action.

Set the `saveQuote` parameter to `false`. It doesn't save the quote and returns an updated context with the eligibility details.

4. Renew Policy—Use the retrieved context to create a new policy record for the renewal term either by:

- Making a POST request on the [Renew Insurance Policy](#) Connect API, or
- Using the [Renew Insurance Policy Action](#) invocable action.

You can change or confirm the policy name, policy number, and effective start and end dates for the renewed policy directly during the API call. Additionally, you can also specify a name for the transaction record that's created along with the renewal record. This transaction record captures the financial details or the audit information related to the renewal record.

Expected Results:

- A brand new policy record is created in Salesforce, representing the coverage for the new policy term. The new policy record includes the details, such as any changed terms, coverages, prices, and the new start and end dates.
- A transaction record is created and linked to the new policy.

Cancel Policy

The policy cancellation process addresses the termination of an insurance policy before its scheduled expiration date. Common reasons for canceling a policy include not paying the premiums or a direct cancellation request from the policyholder.

Make sure you have these details.

- Policy ID of the active insurance policy to be canceled.
- Effective date of cancellation. This date determines when coverage ceases and is a required input for the policy cancellation API.

1. Use the policy ID to cancel the policy either by:
 - Making a POST request on the [Cancel Insurance Policy](#) Connect API, or
 - Using the [Cancel Insurance Policy](#) invocable action.

You can specify a name for the transaction record that's created as part of the cancellation process. The transaction record is used for tracking financial implications, such as refunds.

This process returns the details of the policy versions that are canceled, which include the current version and any future-dated versions. Details of any applicable refunds, such as premium refunds and tax refunds, calculated by the system based on the cancellation date and policy terms are also returned by the API.

Expected Results:

- The specified policy is marked as `Canceled` in Salesforce.
- The policy's effective-to-date field is updated to the day immediately preceding the effective date of cancellation that's specified during the API call.
- Any future-dated policy versions associated with the canceled policy are also automatically canceled.
- Information regarding applicable premium and tax refunds are shown in the API response, which can then be used for downstream financial processing.
- A transaction record is created and linked to the cancellation event.

CHAPTER 5 Insurance Brokerage

In this chapter ...

- [Insurance Brokerage Business APIs](#)
- [Insurance Brokerage Standard Invocable Actions](#)
- [Insurance Brokerage Metadata API Types](#)
- [Insurance Brokerage Apex Reference](#)

Use the Insurance Brokerages developer resources to automate processes related to Policy and Commissions Management. Invoke policy lifecycle services such as Renew, Repurpose, and Cancel using context service. Manage plan benefits using a flexible service layer and Product Catalog.

EDITIONS

Available in: Lightning Experience

Available in: **Enterprise**, **Unlimited**, and **Developer** Editions

Insurance Brokerage Business APIs

Use the Insurance Brokerage Business APIs to manage policy lifecycle, brokerage commissions, and plan benefits.

This table lists the available Insurance Brokerage resources.

Resource	Description
/connect/insurance/brokerage/policies/bulk-cancel (POST)	Bulk cancel policies with the same (existing) carriers for specified insurance policy IDs.
/connect/insurance/brokerage/policies/bulk-renew (POST)	Bulk renew policies with the same (existing) carriers for specified insurance policy IDs.
/connect/insurance/brokerage/commission/process (POST)	Process commission statements in an asynchronous manner using a preconfigured batch job.
/connect/insurance/brokerage/policies/policyId/expected-rates (PATCH)	Calculate the premium, fee, or premium-equivalent amount at InsuranceRatePlanLineItem based on the underlying rate plan.
/connect/insurance/brokerage/policies/policyId/expected-revenue (PATCH)	Calculate the expected revenue, which is derived both from an Insurance Rate Plan Commission object, such as a flat fee and percentage premium, and the premium, fee, or premium equivalent amount on Insurance Rate Plan.
/connect/insurance/brokerage/coverages/coverageId/benefits (POST)	Get, create, update and delete Insurance Policy Coverage Networks and Insurance Policy Coverage Benefits of an Insurance Policy Coverage.
/connect/insurance/brokerage/coverages/coverageId/benefits/product (GET)	Get the product model details along with child components and their attributes for a given insurance policy coverage.
/connect/insurance/brokerage/policies/policyId/renew (POST)	Renew policy for the specified insurance policy ID.

Resources

Learn about the available Insurance Brokerage API resources.

[Requests](#)

Learn about the available Insurance Brokerage API requests.

[Responses](#)

Learn about the response bodies of Insurance Brokerage APIs.

Resources

Learn about the available Insurance Brokerage API resources.

[Bulk Cancel Policy \(POST\)](#)

Bulk cancel policies with the same (existing) carriers for the specified insurance policy IDs.

[Bulk Renew Policy \(POST\)](#)

Bulk renew policies with the same (existing) carriers for the specified insurance policy IDs.

[Commission Processing \(POST\)](#)

Process commission statements in an asynchronous manner using a preconfigured batch job.

[Cost Calculation \(PATCH\)](#)

Calculate the premium, fee, or premium-equivalent amount at InsuranceRatePlanLineItem based on the underlying rate plan. The InsuranceRatePlanLineItem premium, fees, or premium equivalent amount is rolled up to the Insurance Rate Plan, Insurance Coverage, and Insurance Policy Premium Amount.

[Expected Revenue \(PATCH\)](#)

Calculate the expected revenue, derived both from an Insurance Rate Plan Commission object, such as a flat fee and percentage premium, and the premium, fee, or premium equivalent amount on Insurance Rate Plan. The expected revenue amount is rolled up from Insurance Rate Plan to Insurance Coverage and Insurance Policy expected revenue amount.

[Plan Benefits \(GET, POST, PATCH, DELETE\)](#)

Get, create, update and delete Insurance Policy Coverage Networks and Insurance Policy Coverage Benefits of an Insurance Policy Coverage.

[Plan Benefits Product Model \(GET\)](#)

Get the product model details along with child components and their attributes for a given insurance policy coverage.

[Renew Policy \(POST\)](#)

Renew policy for the specified insurance policy ID.

Bulk Cancel Policy (POST)

Bulk cancel policies with the same (existing) carriers for the specified insurance policy IDs.

Resource

```
/connect/insurance/brokerage/policies/bulk-cancel
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/brokerage/policies/bulk-cancel
```

Available version

64.0

HTTP methods

POST

Request body for POST**JSON example**

```
{
  "insurancePolicyIds" : ["0YTxx0000000001GAA", "0YTxx0000000001GAB"],
  "fieldValueMap": { "Name": "test" }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
fieldValueMap	Object	JSON object that contains name-value pairs to store details of insurance policy. The fieldValueMap expect the keys to be API name and is case sensitive.	Required	64.0
insurancePolicyIds	Object	List of insurance policy IDs that must be renewed from the same carrier.	Required	64.0

Response body for POST[Insurance Policy Bulk](#)**Bulk Renew Policy (POST)**

Bulk renew policies with the same (existing) carriers for the specified insurance policy IDs.

Resource

```
/connect/insurance/brokerage/policies/bulk-renew
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/brokerage/policies/bulk-renew
```

Available version

64.0

HTTP methods

POST

Request body for POST**JSON example**

```
{
  "insurancePolicyIds" : ["0YTxx0000000001GAA", "0YTxx0000000001GAB"],
  "fieldValueMap": { "Name": "test" }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
fieldValueMap	Object	JSON object that contains name-value pairs to store details of insurance policy. The fieldValueMap expect the keys to be API name and is case sensitive.	Required	64.0
insurancePolicyIds	Object	List of insurance policy IDs that must be renewed from the same carrier.	Required	64.0

Response body for POST
[Insurance Policy Bulk](#)
Commission Processing (POST)

Process commission statements in an asynchronous manner using a preconfigured batch job.

Resource

```
/connect/insurance/brokerage/commission/process
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/brokerage/commission/process
```

Available version

63.0

HTTP methods

POST

Request body for POST**JSON example**

```
{
  "commissionStatementIds": [
    "199TC0000003kxZYAQ"
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
commissionStatementIds	Object	List of commission statement IDs to be processed.	Required	63.0

Response body for POST
[Insurance Commission Processing](#)

Cost Calculation (PATCH)

Calculate the premium, fee, or premium-equivalent amount at InsuranceRatePlanLineItem based on the underlying rate plan. The InsuranceRatePlanLineItem premium, fees, or premium equivalent amount is rolled up to the Insurance Rate Plan, Insurance Coverage, and Insurance Policy Premium Amount.

Resource

/connect/insurance/brokerage/policies/**policyId**/expected-rates

Resource example

<https://yourinsure.safefore.com/services/ta/v4.0/onet/insure/idea/policies/0733000000944/patchates?insureObjectType=InsurePctPlan&code=HC300000050A>

Available version

63.0

HTTP methods

PATCH

Path parameter for PATCH

Name	Type	Description	Required or Optional	Available Version
policyId	String	ID of parent policy for which the expected cost is calculated.	Required	63.0

Request parameters for PATCH

Parameter Name	Type	Description	Required or Optional	Available Version
businessObjectType	String	Name of business SObject to calculate premium. Possible values are: <ul style="list-style-type: none">InsurancePolicyInsurancePolicyCoverageInsuranceRatePlanInsuranceRatePlanLineItem	Optional	63.0
recordId	String	ID of business SObject for which premium calculation is done.	Optional	63.0

Request body for PATCH

JSON example

```
{
  "action": "rollup",
  "expressionSetName": "expressionSetName",
  "ratePlanLineItems": [
    {
```

```

    "id": "xxxxxx",
    "rate": 100,
    "enrollmentCount": 1000,
    "custom_field_1": "xxxxxx"
  },
  {
    "id": "yyyyy",
    "rate": 50,
    "enrollmentCount": 200,
    "custom_field_1": "xxxxxx"
  }
]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
action	String	Course of action for processing RatePlanLineItems. Valid values: <ul style="list-style-type: none"> rollup: Calculates premium and roll ups it to Insurance Policy level. calculate: Calculates premium and returns it in the output. 	Optional	63.0
expressionSetName	String	Name of the expression set for cost calculation. If not specified, the value from the InsPolicyManagementConfig setup object is taken.	Optional	63.0
ratePlanLineItems	List<Map<String, Object>>	List of insurance rate plan line items.	Optional	63.0

Response body for PATCH

[Cost Calculation](#)

Expected Revenue (PATCH)

Calculate the expected revenue, derived both from an Insurance Rate Plan Commission object, such as a flat fee and percentage premium, and the premium, fee, or premium equivalent amount on Insurance Rate Plan. The expected revenue amount is rolled up from Insurance Rate Plan to Insurance Coverage and Insurance Policy expected revenue amount.

Resource

```
/connect/insurance/brokerage/policies/policyId/expected-revenue
```

Resource example

<https://yourinsurancesfore.com/services/ta/6.0/omr/insure/idea/policies/01SG00000944/extra/extraBusinessObject/InsuredPlan/01SG0000060A>

Available version

63.0

HTTP methods

PATCH

Path parameter for PATCH

Name	Type	Description	Required or Optional	Available Version
policyId	String	ID of parent policy for which the expected revenue is calculated.	Required	63.0

Request parameters for PATCH

Parameter Name	Type	Description	Required or Optional	Available Version
businessObjectType	String	Name of the business sObject to calculate expected revenue. Possible values are: <ul style="list-style-type: none">InsurancePolicyInsurancePolicyCoverageInsuranceRatePlan	Optional	63.0
recordId	String	ID of the business sObject for which expected revenue must be calculated.	Optional	63.0

Request body for PATCH

JSON example

```
{
  "expressionSetName": "expressionSetName",
  "ratePlanCommissions": [
    {
      "Name": "RPC-1",
      "Type": "GradedPercent",
      "amountStartRange": 0,
      "amountEndRange": 1000
    },
    {
      "Name": "RPC-1",
      "Type": "GradedPercent",
      "amountStartRange": 1001,
```

```
        "amountEndRange": 100000
      }
    ]
  }
```

Properties

Name	Type	Description	Required or Optional	Available Version
<code>expressionSetName</code>	String	Name of expression set to be used for expected revenue calculation. If unspecified, a value from the <code>InsPolicyManagementConfig</code> setup object is taken.	Optional	63.0
<code>ratePlanCommissions</code>	List	List of rate plan commissions to be used for expected revenue calculation. This is a valid input only when <code>businessSObjectType</code> is <code>InsuranceRatePlan</code> .	Optional	63.0

Response body for PATCH

[Expected Revenue](#)

Plan Benefits (GET, POST, PATCH, DELETE)

Get, create, update and delete Insurance Policy Coverage Networks and Insurance Policy Coverage Benefits of an Insurance Policy Coverage.

Resource

```
/connect/insurance/brokerage/coverages/coverageId/benefits
```

Available version

63.0

HTTP methods

GET, POST, PATCH, DELETE

Response body for GET

[PlanBenefits](#)

Request body for POST

JSON example

```
{
  "networks": [
    {
      "network": "First Network Tier",
      "name": "Preferred Network"
    },
    {
      "network": "Second Network Tier",
```



```

        "name": "In Network"
      }
    ],
    "benefits": [
      {
        "productId": "01tSG000001lyBFxxY",
        "productCompGgpId": "0y7SG0000000ZiXxxU",
        "productName": "Chiropractic Care",
        "network": "First Network Tier",
        "attributes": [
          {
            "attributeDefinitionId": "0tjSG0000001embxxA",
            "name": "Copay",
            "value": 311
          },
          {
            "attributeDefinitionId": "0tjSG0000001YmMxxU",
            "name": "Deductible",
            "value": "100"
          }
        ]
      }
    ]
  }
}

```

Properties

Name	Type	Descriptions	Required or Optional	Available Version
attributes	Benefit Attributes on page 140	List of benefit attributes.	Optional	63.0
network	String	The tier type of the network.	Optional	63.0
networks	Networks on page 143	Insurance Policy Coverage Networks for creating plan benefits.	Required	63.0
productCompGgpId	String	ID of product component group.	Optional	63.0
productId	String	ID of the product.	Required	63.0
productName	String	Name of the product.	Optional	63.0

Response body for POST

[PlanBenefitsUpsert](#) on page 165

Request body for PATCH

JSON example

```

{
  "networks": [
    {
      "networkId": "1HXSG0000002xxxxAM",

```

```

        "name": "Preferred Network Updated"
      },
      {
        "networkId": "1HXSG0000002xxxxA2",
        "name": "In Network Updated"
      },
      {
        "network": "Third Network Tier",
        "name": "Out Network"
      }
    ],
    "benefits": [
      {
        "benefitId": "1HSSG0000005Ws7xxE",
        "productId": "01tSG0000011yBFxxY",
        "productCompGgpId": "0y7SG0000000ZiXxxU",
        "productName": "Chiropractic Care",
        "attributes": [
          {
            "covBnftAttrId": "1HYSG00000002GPxxY",
            "attributeDefinitionId": "0tjSG0000001embxxA",
            "name": "Copay",
            "value": 311
          },
          {
            "covBnftAttrId": "1HYSG00000002GQxxY",
            "attributeDefinitionId": "0tjSG0000001YmMxxU",
            "name": "Notes",
            "value": "Preferred network limitations apply."
          }
        ]
      }
    ]
  }
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
benefits	Benefits Input []	The benefits of the Insurance Plan Coverage.	Optional	63.0
networks	Networks Input []	The list of networks.	Optional	63.0

Response body for PATCH

[PlanBenefitsUpsert](#)

Request parameters for DELETE

Parameter Name	Type	Description	Required or Optional	Available Version
<code>benefitIds</code>	String[]	ID of Insurance Policy Coverage Benefit. If <code>benefitIds</code> are provided, the service will only delete the respective Insurance Policy Coverage Benefit records.	Optional	63.0
<code>networkIds</code>	String[]	ID of the Insurance Policy Coverage Network. If <code>networkIds</code> are provided, the service will only delete the respective Insurance Policy Coverage Network records and their associated Insurance Policy Coverage Benefit records.	Optional	63.0

Plan Benefits Product Model (GET)

Get the product model details along with child components and their attributes for a given insurance policy coverage.



Note: See [Prerequisites](#) to define the product model structure. If this model isn't followed, the API throws an error.

Resource

```
/connect/insurance/brokerage/coverages/coverageId/benefits/product
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/brokerage/coverages/coverageId/benefits/product
```

Available version

63.0

HTTP methods

GET

Path parameter for GET

Name	Type	Description	Required or Optional	Available Version
<code>coverageId</code>	String	ID of insurance policy coverage.	Required	63.0

Response body for GET

[Plan Benefits Product Model](#)

Renew Policy (POST)

Renew policy for the specified insurance policy ID.

Resource

```
/connect/insurance/brokerage/policies/policyId/renew
```

Resource example

```
https://yourInstance.salesforce.com/services/data/v64.0/connect/insurance/brokerage/policies/policyId/renew
```

Available version

63.0

HTTP methods

POST

Request body for POST**JSON example**

```
{
  "fieldValueMap": { "Name": "demo" },
  "sameCarrier": true
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
fieldValueMap	Object	JSON object that contains name-value pairs that store details of insurance policy.	Required	63.0
sameCarrier	Boolean	Indicates whether the insurance policy is renewed with the same carrier (<code>true</code>) or not (<code>false</code>). If a policy is renewed with the same carrier, the <code>PriorPolicyId</code> field in the new policy is updated with the current policy ID. If not, the <code>PriorPolicyId</code> field is left blank.	Required	63.0

Response body for POST[Insurance Policy Process](#)

Requests

Learn about the available Insurance Brokerage API requests.

[Cost Calculation Input](#)

Input representation of the request to calculate brokerage cost.

[Expected Revenue Input](#)

Input representation of the request to calculate expected brokerage revenue.

[Insurance Commission Processing Input](#)

Input representation details of insurance commission processing.

[Insurance Policy Bulk Input](#)

Input representation details of the request to bulk renew policies. Send the list of insurance policy IDs and field value map of insurance policy fields for bulk policies renewal.

[Insurance Policy Renew Input](#)

Input representation of the details of the request to renew policy.

[Plan Benefit Attribute Input](#)

Represents details of the plan benefits attributes for an Insurance Policy Coverage.

[Plan Benefit Input](#)

Represents details of the plan benefits for an Insurance Policy Coverage.

[Plan Benefit Update Input](#)

Represents details of the plan benefits for an Insurance Policy Coverage.

[Networks Input](#)

Represents Insurance Policy Coverage Networks for creating plan benefits.

[Update Plan Benefit Input](#)

Update Insurance Policy Coverage Networks and Insurance Policy Coverage Benefits by using the ID of an Insurance Policy Coverage.

Cost Calculation Input

Input representation of the request to calculate brokerage cost.

JSON example

```
{
  "action": "rollup",
  "expressionSetName": "expressionSetName",
  "ratePlanLineItems": [
    {
      "id": "xxxxxx",
      "rate": 100,
      "enrollmentCount": 1000,
      "custom_field_1": "xxxxxx"
    },
    {
      "id": "yyyyy",
      "rate": 50,
      "enrollmentCount": 200,
      "custom_field_1": "xxxxxx"
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
action	String	Course of action for processing RatePlanLineItems. Valid values: <ul style="list-style-type: none"> rollup: Calculates premium and roll ups it to Insurance Policy level. calculate: Calculates premium and returns it in the output. 	Optional	63.0
expressionSetName	String	Name of the expression set for cost calculation. If not specified, the value from the InsPolicyManagementConfig setup object is taken.	Optional	63.0
ratePlanLineItems	List<Map<String, Object>>	List of insurance rate plan line items.	Optional	63.0

Expected Revenue Input

Input representation of the request to calculate expected brokerage revenue.

JSON example

```
{
  "expressionSetName": "expressionSetName",
  "ratePlanCommissions": [
    {
      "Name": "RPC-1",
      "Type": "GradedPercent",
      "amountStartRange": 0,
      "amountEndRange": 1000
    },
    {
      "Name": "RPC-1",
      "Type": "GradedPercent",
      "amountStartRange": 1001,
      "amountEndRange": 100000
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
expressionSetName	String	Name of expression set to be used for expected revenue calculation. If	Optional	63.0

Name	Type	Description	Required or Optional	Available Version
		unspecified, a value from the <code>InsPolicyManagementConfig</code> setup object is taken.		
<code>ratePlanCommissions</code>	List	List of rate plan commissions to be used for expected revenue calculation. This is a valid input only when <code>businessSObjectType</code> is <code>InsuranceRatePlan</code> .	Optional	63.0

Insurance Commission Processing Input

Input representation details of insurance commission processing.

JSON example

```
{
  "commissionStatementIds": [
    "199TC0000003kxZYAQ"
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
<code>commissionStatementIds</code>	Object	List of commission statement IDs to be processed.	Required	63.0

Insurance Policy Bulk Input

Input representation details of the request to bulk renew policies. Send the list of insurance policy IDs and field value map of insurance policy fields for bulk policies renewal.

JSON example

```
{
  "insurancePolicyIds" : ["0YTx0000000001GAA", "0YTx0000000001GAB"],
  "fieldValueMap": { "Name": "test" }
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
<code>fieldValueMap</code>	Object	JSON object that contains name-value pairs to store details of insurance policy.	Required	64.0

Name	Type	Description	Required or Optional	Available Version
		The fieldValueMap expect the keys to be API name and is case sensitive.		
insurancePolicyIds	Object	List of insurance policy IDs that must be renewed from the same carrier.	Required	64.0

Insurance Policy Renew Input

Input representation of the details of the request to renew policy.

JSON example

```
{
  "fieldValueMap": { "Name": "demo" },
  "sameCarrier": true
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
fieldValueMap	Object	JSON object that contains name-value pairs that store details of insurance policy.	Required	63.0
sameCarrier	Boolean	Indicates whether the insurance policy is renewed with the same carrier (<code>true</code>) or not (<code>false</code>). If a policy is renewed with the same carrier, the <code>PriorPolicyId</code> field in the new policy is updated with the current policy ID. If not, the <code>PriorPolicyId</code> field is left blank.	Required	63.0

Plan Benefit Attribute Input

Represents details of the plan benefits attributes for an Insurance Policy Coverage.

JSON example

```
{
  "attributes": [
    {
      "attributeDefinitionId": "0tjSG0000001embxxA",
      "name": "Copay",
      "value": 400
    },
    {
      "attributeDefinitionId": "0tjSG00000020yPxxQ",
      "name": "Deductible waiver",
      "value": "Deductible waived"
    }
  ]
}
```



```

    }
  ]
}

```

Properties

Name	Type	Description	Required or Optional	Available Version
attributeDefinitionId	String	ID of the Attribute Definition.	Required	63.0
name	String	Name of the attribute.	Optional	63.0
value	String	Value of the attribute.	Required	63.0

Plan Benefit Input

Represents details of the plan benefits for an Insurance Policy Coverage.

JSON example

```

{
  "networks": [
    {
      "network": "First Network Tier",
      "name": "Preferred Network"
    },
    {
      "network": "Second Network Tier",
      "name": "In Network"
    }
  ],
  "benefits": [
    {
      "productId": "01tSG000001lyBFxxY",
      "productCompGgpId": "0y7SG0000000ZiXxxU",
      "productName": "Chiropractic Care",
      "network": "First Network Tier",
      "attributes": [
        {
          "attributeDefinitionId": "0tjSG00000001embxxA",
          "name": "Copay",
          "value": 311
        },
        {
          "attributeDefinitionId": "0tjSG00000001YmMxxU",
          "name": "Deductible",
          "value": "100"
        }
      ]
    }
  ]
}

```

Properties

Name	Type	Descriptions	Required or Optional	Available Version
attributes	Benefit Attributes on page 140	List of benefit attributes.	Optional	63.0
network	String	The tier type of the network.	Optional	63.0
networks	Networks on page 143	Insurance Policy Coverage Networks for creating plan benefits.	Required	63.0
productCompGgpId	String	ID of product component group.	Optional	63.0
productId	String	ID of the product.	Required	63.0
productName	String	Name of the product.	Optional	63.0

Plan Benefit Update Input

Represents details of the plan benefits for an Insurance Policy Coverage.

JSON example

```
{
  "benefits": [
    {
      "benefitId": "1HSSG0000005Ws7xxE",
      "productId": "01tSG0000011yBFxxY",
      "productCompGgpId": "0y7SG0000000ZiXxxU",
      "productName": "Chiropractic Care",
      "attributes": [
        {
          "covBnftAttrId": "1HYSG00000002GPxxY",
          "attributeDefinitionId": "0tjSG00000001embxxA",
          "name": "Copay",
          "value": 311
        },
        {
          "covBnftAttrId": "1HYSG00000002GQxxY",
          "attributeDefinitionId": "0tjSG00000001YmMxxU",
          "name": "Notes",
          "value": "Preferred network limitations apply."
        }
      ]
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
benefitId	String	ID of the Insurance Policy Coverage Benefit.	Required	63.0
productId	String	ID of the product.	Optional	63.0
productCompGgpId	String	ID of product component group	Optional	63.0
productName	String	Name of the product.	Optional	63.0
network	String	The tier type of the network.	Optional	63.0
attributes	String	List of attributes.	Optional	63.0
covBnftAttrId	String	ID of Insurance Policy Coverage Benefit Attribute.	Required	63.0
name	String	Name of the attribute.	Optional	63.0
value	String	Value of the attribute.	Optional	63.0
attributeDefinitionId	String	ID of the Attribute Definition.	Optional	63.0

Networks Input

Represents Insurance Policy Coverage Networks for creating plan benefits.

JSON example

```
{
  "networks": [
    {
      "network": "First Network Tier",
      "name": "Preferred Network Updated"
    },
    {
      "network": "Second Network Tier",
      "name": "In Network Updated"
    },
    {
      "network": "Third Network Tier",
      "name": "Out Network"
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
name	String	The name of the network.	Required	63.0

Name	Type	Description	Required or Optional	Available Version
network	String	Tier type of the network.	Required	63.0

Update Plan Benefit Input

Update Insurance Policy Coverage Networks and Insurance Policy Coverage Benefits by using the ID of an Insurance Policy Coverage.

JSON example

```
{
  "networks": [
    {
      "networkId": "1HXSG0000002xxxxAM",
      "name": "Preferred Network Updated"
    },
    {
      "networkId": "1HXSG0000002xxxxA2",
      "name": "In Network Updated"
    },
    {
      "network": "Third Network Tier",
      "name": "Out Network"
    }
  ],
  "benefits": [
    {
      "benefitId": "1HSSG0000005Ws7xxE",
      "productId": "01tSG000001lyBFxxY",
      "productCompGgpId": "0y7SG0000000ZiXxxU",
      "productName": "Chiropractic Care",
      "attributes": [
        {
          "covBnftAttrId": "1HYSG00000002GPxxY",
          "attributeDefinitionId": "0tjSG0000001embxxA",
          "name": "Copay",
          "value": 311
        },
        {
          "covBnftAttrId": "1HYSG00000002GQxxY",
          "attributeDefinitionId": "0tjSG0000001YmMxxU",
          "name": "Notes",
          "value": "Preferred network limitations apply."
        }
      ]
    }
  ]
}
```

Properties

Name	Type	Description	Required or Optional	Available Version
benefits	Benefits Input []	The benefits of the Insurance Plan Coverage.	Optional	63.0
networks	Networks Input []	The list of networks.	Optional	63.0

Responses

Learn about the response bodies of Insurance Brokerage APIs.

[Amount with Frequency](#)

Output representation of the amount details with frequency.

[Attribute Detail](#)

Output representation of product attribute details.

[Attribute Picklist Value](#)

Output representation of the picklist value details.

[Benefits](#)

Output representation details of the plan benefits for an insurance policy coverage plan.

[Cost Calculation](#)

Output representation details of the cost calculation.

[Expected Revenue Coverage](#)

Output representation details of the coverage and associated expected revenue.

[Expected Revenue Policy](#)

Output representation of the associated expected revenue details for a policy, such as coverages and rate plans.

[Expected Revenue](#)

Output representation of the details of the expected revenue.

[Commission Processing Details](#)

Output representation details of the commission processing.

[Insurance Policy Bulk](#)

Output representation details of bulk renew policy and bulk cancel policy requests.

[Insurance Policy Process](#)

Output representation details of the renewed policy.

[Plan Benefits Product Model Details](#)

Output representation of the product model details and associated product components.

[Rate Plan Commission](#)

Output representation details of the rate plan commission.

[Rate Plan](#)

Output representation of the details of the rate plans associated directly with the policy or through coverages.

Product

Output representation of product details.

Plan Benefits Attribute

Represents details of the plan benefits attributes for an Insurance Policy Coverage.

Plan Benefits Attributes

Output representation details of the benefits attributes for an insurance policy coverage plan.

Plan Benefit Networks

Output representation details of the insurance policy coverage networks.

Plan Benefits

Output representation details of the create and update plan benefits request.

Plan Benefits Details

Output representation details of the insurance policy benefits.

Product

Output representation of product details.

Amount with Frequency

Output representation of the amount details with frequency.

JSON example

This example shows a sample success response.

```
"amountWithFrequency": {
  "annually": 1000.3,
  "monthly": 100.03,
  "quarterly": 300.09
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
annually	Double	Expected revenue amount that's calculated annually.	Small, 63.0	63.0
monthly	Double	Expected revenue amount that's calculated monthly.	Small, 63.0	63.0
quarterly	Double	Expected revenue amount that's calculated quarterly.	Small, 63.0	63.0

Attribute Detail

Output representation of product attribute details.

JSON example

This request shows a sample success response.

```
"attributes": [
  {
    "id": "0tjSG0000001embYAA",
    "sequence": 1, #optional
    "name": "Copay",
    "label": "Copay Currency", #optional
    "dataType": "CURRENCY",
    "keyHidden": false,
    "isActive": true
  },
  {
    "id": "0tjSG0000001eoDYAQ",
    "sequence": 1,
    "name": "Coinsurance",
    "label": "Coinsurance Percentage",
    "dataType": "PERCENTAGE",
    "keyHidden": false,
    "isActive": true
  }
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
attributeDefinitionId	String	ID of the attribute definition.	Small, 63.0	63.0
attributePicklistValueId	String	ID of the attribute picklist value.	Small, 63.0	63.0
dataType	String	Data type of attribute, such as Percentage.	Small, 63.0	63.0
id	String	ID of the product.	Small, 63.0	63.0
isActive	Boolean	Indicates whether the attribute is active (true) or not (false).	Small, 63.0	63.0
keyHidden	Boolean	Indicates whether the attribute key is hidden (true) or (not).	Small, 63.0	63.0
label	String	Label of the attribute.	Small, 63.0	63.0
name	String	Name of the attribute.	Small, 63.0	63.0
picklistValues	Attribute Picklist Value[]	List of picklist values.	Small, 63.0	63.0
sequence	Integer	Sequence of attribute.	Small, 63.0	63.0
value	String	Value of attribute.	Small, 63.0	63.0

Attribute Picklist Value

Output representation of the picklist value details.

Property Name	Type	Description	Filter Group and Version	Available Version
attributePicklistId	String	ID of the attribute picklist.	Small, 63.0	63.0
id	String	ID of the product.	Small, 63.0	63.0
name	String	Name of the attribute.	Small, 63.0	63.0
value	String	Value of the attribute.	Small, 63.0	63.0

Benefits

Output representation details of the plan benefits for an insurance policy coverage plan.

JSON example

```
{
  "benefits": [
    {
      "benefitId": "bft1",
      "benefitSequence": 1,
      "productId": "0ltSG000001lyBFxxY",
      "productName": "Preventive Care",
      "productCompGrpId": "0y7SG0000000Q8zxxE",
      "productCompGgpName": "General plan information",
      "productCompGgpSequence": 1,
      "network": "First Network Tier",
      "attributes": [
        {
          "attributeDefinitionId": "0tjSG0000001embxxA",
          "name": "Copay",
          "value": 400,
          "sequence": 1,
          "keyHidden": false,
          "dataType": "Currency"
        },
        {
          "attributeDefinitionId": "0tjSG00000020yPxxQ",
          "name": "Deductible waiver",
          "value": "Deductible waived",
          "sequence": 2,
          "keyHidden": true,
          "dataType": "Text"
        },
        {
          "attributeDefinitionId": "0tjSG0000001YmMxxU",
          "name": "Notes",
          "value": "In network limitations apply.",
          "sequence": 2,
          "keyHidden": true,

```



```

        "dataType": "Text"
      }
    ]
  }
}

```

Properties

Name	Type	Description	Available Version
benefitId	String	ID of the Insurance Policy Coverage Benefit.	63.0
benefitSequence	String	Field sequence of the Insurance Policy Coverage Benefit.	63.0
network	String	Type of tier of the network.	63.0
productId	String	ID of the product.	63.0
productCompGgpId	String	ID of product component group.	63.0
productCompGgpName	String	Name of product component group.	63.0
productCompGgpSequence	String	Field sequence of the product component group.	63.0
productName	String	Name of the product.	63.0
attributes	Attributes on page 163	List of attributes of the plan benefits.	63.0

Cost Calculation

Output representation details of the cost calculation.

JSON example

Here's a sample success response.

```

{
  "isSuccess": true,
  "errors": [],
  "policy": {
    "id": "p111",
    "premiumAmountWithFrequency": {
      "Annually": "xxx.xx",
      "Monthly": "xxx.xxx",
      "Quarterly": "xx.xxx"
    },
    "premiumEqvAmountWithFrequency": {
      "Annually": "xx.xx",
      "Monthly": "xx.xx",
      "Quarterly": "xxx.xxx"
    },
    "feeAmountWithFrequency": {
      "Annually": "xx.xx",
      "Monthly": "xx.xx",
      "Quarterly": "xxx.xxx"
    }
  }
}

```

```

    },
    "coverages": [
      {
        "id": "pc111",
        "premiumAmountWithFrequency": {
          "Annually": "xxx.xx",
          "Monthly": "xxx.xxx",
          "Quarterly": "xx.xxx"
        },
        "premiumEqvAmountWithFrequency": {
          "Annually": "xx.xx",
          "Monthly": "xx.xx",
          "Quarterly": "xxx.xxx"
        },
        "feeAmountWithFrequency": {
          "Annually": "xx.xx",
          "Monthly": "xx.xx",
          "Quarterly": "xxx.xxx"
        }
      },
      "ratePlans": [
        {
          "id": "rp11",
          "totalPremiumAmount": 350,
          "totalFeeAmount": "xyz",
          "totalPremiumEqvAmount": "xyz",
          "ratePlanLineItems": [
            {
              "id": "rpl1111",
              "premiumAmount": 100
            },
            {
              "id": "rpl1112",
              "premiumAmount": 150
            }
          ]
        },
        {
          "id": "rp12",
          "totalPremiumAmount": 580,
          "ratePlanLineItems": [
            {
              "id": "rpl1121",
              "premiumAmount": 80
            },
            {
              "id": "rpl1122",
              "premiumAmount": 200
            },
            {
              "id": "rpl1123",
              "premiumAmount": 300
            }
          ]
        }
      ]
    }
  ]
}

```

```

    ]
  }
}
}

```

This request shows a sample error response.

```

{
  "isSuccess":true,
  "errors" : [
    {
      "code":INVALID_INPUT,
      "message": "Specify a valid ratePlanId."
    }
  ]
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
isSuccess	Boolean	Indicates whether the API request is successful (<code>true</code>) or not (<code>false</code>).	Small, 63.0	63.0
policy	Map<String, Object>	Map of policy and cost calculation details from <code>policy</code> to <code>ratePlanLineItems</code> .	Small, 63.0	63.0

Expected Revenue Coverage

Output representation details of the coverage and associated expected revenue.

JSON example

This example shows a sample success response.

```

"coverages": [
  {
    "id": "pc111",
    "amountWithFrequency": {
      "annually": 1000.3,
      "monthly": 100.03,
      "quarterly": 300.09
    },
    "ratePlans": [
      {
        "id": "rp11",
        "totalPremiumAmount": 350,
        "totalFeeAmount": 200,
        "totalPremiumEqvAmount": 100,
        "expectedRevenueAmount": 45,

```

```

    "ratePlanCommissions": [
      {
        "id": "rpc111",
        "premiumPercent": 100,
        "type": "FlatPercentOfPremium"
      }
    ]
  },
  {
    "id": "rp12",
    "totalPremiumAmount": 350,
    "totalFeeAmount": 200,
    "totalPremiumEqvAmount": 100,
    "expectedRevenueAmount": 45,
    "ratePlanCommissions": [
      {
        "id": "rpc111",
        "premiumPercent": null,
        "type": "GradedPercent",
        "amountStartRange": 0,
        "amountEndRange": 1000
      },
      {
        "id": "rpc112",
        "premiumPercent": null,
        "type": "GradedPercent",
        "amountStartRange": 1001,
        "amountEndRange": 10000
      }
    ]
  }
]
}
]

```

Property Name	Type	Description	Filter Group and Version	Available Version
amountWithFrequency	Amount With Frequency	Expected revenue amount with frequency.	Small, 63.0	63.0
id	String	ID of the coverage associated with the policy for which the expected revenue is calculated.	Small, 63.0	63.0
ratePlans	Rate Plan	Details of the rate plans associated with the policy or through coverages.	Small, 63.0	63.0

Expected Revenue Policy

Output representation of the associated expected revenue details for a policy, such as coverages and rate plans.

JSON example

This example shows a sample success response.

```
{
  "policy": {
    "id": "p111",
    "amountWithFrequency": {
      "annually": 1000.3,
      "monthly": 100.03,
      "quarterly": 300.09
    },
    "coverages": [
      {
        "id": "pc111",
        "amountWithFrequency": {
          "annually": 1000.3,
          "monthly": 100.03,
          "quarterly": 300.09
        },
        "ratePlans": [
          {
            "id": "rp11",
            "totalPremiumAmount": 350,
            "totalFeeAmount": 200,
            "totalPremiumEqvAmount": 100,
            "expectedRevenueAmount": 45,
            "ratePlanCommissions": [
              {
                "id": "rpc111",
                "premiumPercent": 100,
                "type": "FlatPercentOfPremium"
              }
            ]
          }
        ],
      },
      {
        "id": "rp12",
        "totalPremiumAmount": 350,
        "totalFeeAmount": 200,
        "totalPremiumEqvAmount": 100,
        "expectedRevenueAmount": 45,
        "ratePlanCommissions": [
          {
            "id": "rpc111",
            "premiumPercent": null,
            "type": "GradedPercent",
            "amountStartRange": 0,
            "amountEndRange": 1000
          },
          {
            "id": "rpc112",
            "premiumPercent": null,
            "type": "GradedPercent",
            "amountStartRange": 1001,
            "amountEndRange": 10000
          }
        ]
      }
    ]
  }
}
```

```
    },
  ],
},
],
},
],
},
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
amountWithFrequency	Amount With Frequency[]	Expected revenue amount with frequency.	Small, 63.0	63.0
coverages	Expected Rev Coverage[]	Details of policy coverages, such as associated rate plans.	Small, 63.0	63.0
id	String	ID of the policy for which expected revenue is calculated.	Small, 63.0	63.0
ratePlans	Rate Plan[]	Details of rate plans.	Small, 63.0	63.0

Expected Revenue

Output representation of the details of the expected revenue.

JSON example

This example shows a sample success response.

```
{
  "policy": {
    "id": "p111",
    "amountWithFrequency": {
      "annually": 1000.3,
      "monthly": 100.03,
      "quarterly": 300.09
    },
    "coverages": [
      {
        "id": "pc111",
        "amountWithFrequency": {
          "annually": 1000.3,
          "monthly": 100.03,
          "quarterly": 300.09
        },
        "ratePlans": [
          {
            "id": "rp11",
            "totalPremiumAmount": 350,
            "totalFeeAmount": 200,
            "totalPremiumEqvAmount": 100,
            "expectedRevenueAmount": 45,
            "ratePlanCommissions": [
```

```

    {
      "id": "rpc111",
      "premiumPercent": 100,
      "type": "FlatPercentOfPremium"
    }
  ],
},
{
  "id": "rp12",
  "totalPremiumAmount": 350,
  "totalFeeAmount": 200,
  "totalPremiumEqvAmount": 100,
  "expectedRevenueAmount": 45,
  "ratePlanCommissions": [
    {
      "id": "rpc111",
      "premiumPercent": null,
      "type": "GradedPercent",
      "amountStartRange": 0,
      "amountEndRange": 1000
    },
    {
      "id": "rpc112",
      "premiumPercent": null,
      "type": "GradedPercent",
      "amountStartRange": 1001,
      "amountEndRange": 10000
    }
  ],
}
]
}
]
}
}
}
```

This example shows a sample error response.

```

{
  "isSuccess":false,
  "errors" : [
    {
      "code":INVALID_INPUT,
      "message": "Specify a valid ratePlanId."
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response[]	List of errors encountered during the processing of the API request.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
isSuccess	Boolean	Indicates whether the API request is successful (<code>true</code>) or not (<code>false</code>).	Small, 63.0	63.0
policy	Expected Revenue Policy[]	Map of the policy and expected revenue calculation details.	Small, 63.0	63.0

Commission Processing Details

Output representation details of the commission processing.

JSON example

This example shows a sample success response.

```
{
  "status": "Success",
  "message": "The processing has started for the commission statement CS-00000001."
}
```

This example shows a sample error response.

```
{
  "status": "Fail",
  "message": "Processing failed for the commission statement The processing has started for the commission statement CS-00000001 ."
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
message	String	Status message of the API execution.	Small, 63.0	63.0
status	String	Status of the API execution. Valid values are: <ul style="list-style-type: none"> • Success • Fail 	Small, 63.0	63.0

Insurance Policy Bulk

Output representation details of bulk renew policy and bulk cancel policy requests.

JSON example

This example shows a sample success response.

```
{
  "errors": [{ "code": "", "message": "" }],
  "status": "Success",
  "message": "policy renewed successfully."
}
```


This example shows a sample error response.

```
{
  "errors": [{ "code": "", "message": "" }],
  "status": "Fail",
  "message": "error message."
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
errors	Error Response	List of errors encountered during the processing of the API request.	Small, 63.0	63.0
message	String	Status message of the API execution.	Small, 63.0	63.0
status	String	Status of the API execution. Valid values are: <ul style="list-style-type: none"> • Success • Fail 	Small, 63.0	63.0

Insurance Policy Process

Output representation details of the renewed policy.

JSON example

This example shows a sample success response.

```
{
  "errors": [{ "code": "", "message": "" }],
  "status": "Success",
  "policyId": "testId",
  "message": "policy renewed successfully."
}
```

This example shows a sample error response.

```
{
  "errors": [{ "code": "", "message": "" }],
  "status": "Fail",
  "policyId": "",
  "message": "error message."
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
policyId	String	ID of the renewed policy.	Small, 63.0	63.0

Plan Benefits Product Model Details

Output representation of the product model details and associated product components.

JSON example

This example shows a sample success response.

```
{
  "isSuccess":true,
  "errors":[],
  "productId": "01tSG0000013itZYAQ",
  "benefits": [
    {
      "id": "01tSG0000011y9dYAA",
      "name": "Annual deductible/Individual",
      "componentGroupId": "0y7SG0000000Q8zYAE",
      "componentGroupName": "General plan information",
      "componentGroupSequence": 1, #optional
      "sequence": 1, #optional
      "attributes": [
        {
          "id": "0tjSG0000001embYAA",
          "sequence": 1, #optional
          "name": "Copay",
          "label": "Copay Currency", #optional
          "dataType": "CURRENCY",
          "keyHidden":false,
          "isActive": true
        },
        {
          "id": "0tjSG0000001eoDYAQ",
          "sequence": 1,
          "name": "Coinsurance",
          "label": "Coinsurance Percentage",
          "dataType": "PERCENTAGE",
          "keyHidden":false,
          "isActive": true
        }
      ]
    }
  ]
}
```

This request shows a sample error response.

```
{
  "isSuccess":false,
  "errors" : [
    {
      "code":INVALID_INPUT,
      "message": "Specify a valid coverageId."
    }
  ]
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
benefits	Product[]	List of the product component groups and associated attributes.	Small, 63.0	63.0
errors	Error Response[]	List of errors during the processing of the API request.	Small, 63.0	63.0
isSuccess	Boolean	Indicates whether the API request is successful (<code>true</code>) or not (<code>false</code>).	Small, 63.0	63.0
productId	String	ID of the product for which the benefits are fetched.	Small, 63.0	63.0

Rate Plan Commission

Output representation details of the rate plan commission.

JSON example

This example shows a sample success response.

```
[
  {
    "id": "rpc111",
    "premiumPercent": null,
    "type": "GradedPercent",
    "amountStartRange": 0,
    "amountEndRange": 1000
  },
  {
    "id": "rpc112",
    "premiumPercent": null,
    "type": "GradedPercent",
    "amountStartRange": 1001,
    "amountEndRange": 10000
  },
]
```

Property Name	Type	Description	Filter Group and Version	Available Version
amountEndRange	Double	End range of amount for graded percent of premium.	Small, 63.0	63.0
amountStartRange	Double	Start range of amount for graded percent of premium.	Small, 63.0	63.0
fee	Double	Commission fee when flat fee or capitated fee is selected.	Small, 63.0	63.0
feePercent	Double	Percentage of fee earned as commission when flat percentage of fee is selected.	Small, 63.0	63.0

Property Name	Type	Description	Filter Group and Version	Available Version
id	String	ID of the rate plan commission.	Small, 63.0	63.0
numberEndRange	Integer	End range of the grade volume, which represents the employee count.	Small, 63.0	63.0
numberStartRange	Integer	Start range of the grade volume, which represents the employee count.	Small, 63.0	63.0
premiumPercent	Double	Percentage of premium earned as commission when graded percent of premium or flat percent of premium is selected.	Small, 63.0	63.0
type	String	Type of commission structure. Valid values are: <ul style="list-style-type: none"> • Flat Percent of Premium • Flat Fee • Graded Percent of Premium • Graded Volume • Capitated Fee • Flat percent of Fee 	Small, 63.0	63.0

Rate Plan

Output representation of the details of the rate plans associated directly with the policy or through coverages.

JSON example

This example shows a sample success response.

```

"ratePlans": [
  {
    "id": "rp11",
    "totalPremiumAmount": 350,
    "totalFeeAmount": 200,
    "totalPremiumEqvAmount": 100,
    "expectedRevenueAmount": 45,
    "ratePlanCommissions": [
      {
        "id": "rpc111",
        "premiumPercent": 100,
        "type": "FlatPercentOfPremium"
      }
    ]
  },
  {
    "id": "rp12",
    "totalPremiumAmount": 350,

```

```

    "totalFeeAmount": 200,
    "totalPremiumEqvAmount": 100,
    "expectedRevenueAmount": 45,
    "ratePlanCommissions": [
      {
        "id": "rpc111",
        "premiumPercent": null,
        "type": "GradedPercent",
        "amountStartRange": 0,
        "amountEndRange": 1000
      },
      {
        "id": "rpc112",
        "premiumPercent": null,
        "type": "GradedPercent",
        "amountStartRange": 1001,
        "amountEndRange": 10000
      }
    ]
  }
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
expectedRevenue Amount	Double	Expected revenue amount for the specified policy.	Small, 63.0	63.0
id	String	ID of the rate plan.	Small, 63.0	63.0
ratePlan Commissions	Rate Plan Commission[]	Details of the rate plan commissions.	Small, 63.0	63.0
totalFee Amount	Double	Total fee amount that's calculated for rate plan and then rolled up to policy.	Small, 63.0	63.0
totalPremium Amount	Double	Total premium amount that's calculated for rate plan and then rolled up to policy.	Small, 63.0	63.0
totalPremium EqvAmount	Double	Total premium equivalent amount that's calculated for rate plan and then rolled up to policy.	Small, 63.0	63.0

Product

Output representation of product details.

JSON example

This example shows a sample success response.

```

"benefits": [
  {
    "id": "01tSG0000011y9dYAA",

```

```

    "name": "Annual deductible/Individual",
    "componentGroupId": "0y7SG000000Q8zYAE",
    "componentGroupName": "General plan information",
    "componentGroupSequence": 1, #optional
    "sequence": 1, #optional
    "attributes": [
      {
        "id": "0tjSG0000001embYAA",
        "sequence": 1, #optional
        "name": "Copoly",
        "label": "Copoly Currency", #optional
        "dataType": "CURRENCY",
        "keyHidden": false,
        "isActive": true
      },
      {
        "id": "0tjSG0000001eoDYAQ",
        "sequence": 1,
        "name": "Coinsurance",
        "label": "Coinsurance Percentage",
        "dataType": "PERCENTAGE",
        "keyHidden": false,
        "isActive": true
      }
    ]
  }
}

```

Property Name	Type	Description	Filter Group and Version	Available Version
attributes	Attribute Detail[]	List of attributes.	Small, 63.0	63.0
componentGroupId	String	ID of the component group.	Small, 63.0	63.0
componentGroupName	String	Name of the component group.	Small, 63.0	63.0
componentGroupSequence	Integer	Sequence of the component group.	Small, 63.0	63.0
id	String	ID of the product.	Small, 63.0	63.0
name	String	Name of the product.	Small, 63.0	63.0
sequence	Integer	Sequence of the product.	Small, 63.0	63.0

Plan Benefits Attribute

Represents details of the plan benefits attributes for an Insurance Policy Coverage.

JSON example

```

{
  "attributes": [
    {

```

```

        "attributeDefinitionId": "0tjSG0000001embxxA",
        "name": "Copay",
        "value": 400,
        "sequence": 1,
        "keyHidden": false,
        "dataType": "Currency"
      },
      {
        "attributeDefinitionId": "0tjSG00000020yPxxQ",
        "name": "Deductible waiver",
        "value": "Deductible waived",
        "sequence": 2,
        "keyHidden": true,
        "dataType": "Text"
      }
    ]
  }
}

```

Properties

Name	Type	Description	Available Version
attributeDefinitionId	String	ID of the Attribute Definition.	63.0
name	String	Name of the attribute.	63.0
value	String	Value of the attribute.	63.0
sequence	Integer	Sequence field of the product classification attribute or product attribute definition.	63.0
keyHidden	Boolean	IsHidden field of the Product classification attribute or Product attribute definition.	63.0
dataType	String	Data type of the Attribute Definition.	63.0

Plan Benefits Attributes

Output representation details of the benefits attributes for an insurance policy coverage plan.

JSON example

```

{
  "attributes": [
    {
      "attributeDefinitionId": "0tjSG0000001embxxA",
      "name": "Copay",
      "value": 400,
      "sequence": 1,
      "keyHidden": false,
      "dataType": "Currency"
    },
    {
      "attributeDefinitionId": "0tjSG00000020yPxxQ",
      "name": "Deductible waiver",

```

```

    "value": "Deductible waived",
    "sequence": 2,
    "keyHidden": true,
    "dataType": "Text"
  },
  {
    "attributeDefinitionId": "0tjSG0000001YmMxxU",
    "name": "Notes",
    "value": "In network limitations apply.",
    "sequence": 2,
    "keyHidden": true,
    "dataType": "Text"
  }
]
}

```

Properties

Name	Type	Description	Available Version
attributeDefinitionId	String	Indicates whether the request is successful (<code>true</code>) or not (<code>false</code>).	63.0
dataType	String	Datatype of the Attribute Definition.	63.0
keyHidden	String	IsHidden field of the Product classification attribute or Product attribute definition.	63.0
name	String	Type of tier of the network.	63.0
sequence	String	Sequence field of the Product classification attribute or Product attribute definition.	63.0
value	String	ID of the product.	63.0

Plan Benefit Networks

Output representation details of the insurance policy coverage networks.

JSON example

```

{
  "networks": [
    {
      "id": "n1",
      "network": "FIRST_NETWORK_TIER",
      "name": "Preferred Network",
      "sequence": 1
    },
    {
      "id": "n2",
      "network": "SECOND_NETWORK_TIER",
      "name": "In Network",
      "sequence": 2
    }
  ]
}

```



```
]
}
```

Properties

Name	Type	Description	Available Version
id	String	ID of the insurance policy coverage network.	63.0
name	String	Name of the insurance policy coverage network.	63.0
network	String	Insurance policy coverage network tier type.	63.0
sequence	String	Sequence field of the insurance policy coverage network.	63.0

Plan Benefits

Output representation details of the create and update plan benefits request.

JSON example

```
{
  "isSuccess":true,
  "errors":[]
}
```

Properties

Name	Type	Description	Available Version
isSuccess	Boolean	Indicates whether the request is successful (<code>true</code>) or not (<code>false</code>).	63.0
errors	String	The details of any errors.	63.0

Plan Benefits Details

Output representation details of the insurance policy benefits.

JSON example

```
{
  "isSuccess": true,
  "errors": [],
  "networks": [
    {
      "id": "n1",
      "network": "FIRST_NETWORK_TIER",
      "name": "Preferred Network",
      "sequence": 1
    },
    {
      "id": "n2",
      "network": "SECOND_NETWORK_TIER",
```

```

        "name": "In Network",
        "sequence": 2
    }
],
"benefits": [
    {
        "benefitId": "bft1",
        "benefitSequence": 1,
        "productId": "01tSG000001lyBFYAY",
        "productName": "Preventive Care",
        "productCompGrpId": "0y7SG0000000Q8zYAE",
        "productCompGgpName": "General plan information",
        "productCompGgpSequence": 1,
        "network": "First Network Tier",
        "attributes": [
            {
                "attributeDefinitionId": "0tjSG0000001embYAA",
                "name": "Copay",
                "value": 400,
                "sequence": 1,
                "keyHidden": false,
                "dataType": "Currency"
            },
            {
                "attributeDefinitionId": "0tjSG00000020yPYAQ",
                "name": "Deductible waiver",
                "value": "Deductible waived",
                "sequence": 2,
                "keyHidden": true,
                "dataType": "Text"
            },
            {
                "attributeDefinitionId": "0tjSG0000001YmMYAU",
                "name": "Notes",
                "value": "In network limitations apply.",
                "sequence": 2,
                "keyHidden": true,
                "dataType": "Text"
            }
        ],
        "combinedValue": "Copay $400 | Deductible waived | In network limitations apply."
    }
]
}

```

Properties

Name	Type	Description	Available Version
attributes	Benefit Attributes on page 162	List of attributes.	63.0
benefits	Benefits on page 148	Details of the plan benefit.	63.0

Name	Type	Description	Available Version
combinedValue	String	Combined value of the Insurance Policy Coverage Benefit Attribute for a benefit.	63.0
errors	List	List of errors.	63.0
isSuccess	Boolean	Boolean to determine whether it was success or failure.	63.0
network	String	The tier type of the network.	63.0
networks	Networks on page 164	Insurance Policy Coverage Networks for creating plan benefits.	63.0
productCompGgpId	String	ID of product component group	63.0
productId	String	ID of the product.	63.0
productName	String	Name of the product.	63.0

Product

Output representation of product details.

JSON example

This example shows a sample success response.

```

"benefits": [
  {
    "id": "01tSG000001ly9dYAA",
    "name": "Annual deductible/Individual",
    "componentGroupId": "0y7SG0000000Q8zYAE",
    "componentGroupName": "General plan information",
    "componentGroupSequence": 1, #optional
    "sequence": 1, #optional
    "attributes": [
      {
        "id": "0tjSG0000001embYAA",
        "sequence": 1, #optional
        "name": "Copay",
        "label": "Copay Currency", #optional
        "dataType": "CURRENCY",
        "keyHidden": false,
        "isActive": true
      },
      {
        "id": "0tjSG0000001eoDYAQ",
        "sequence": 1,
        "name": "Coinsurance",
        "label": "Coinsurance Percentage",
        "dataType": "PERCENTAGE",
        "keyHidden": false,
        "isActive": true
      }
    ]
  }
]

```

```
{
  "attributes": [
    {
      "name": "id",
      "value": "1234567890"
    },
    {
      "name": "name",
      "value": "ABC Insurance Policy"
    },
    {
      "name": "sequence",
      "value": 1
    }
  ],
  "componentGroupId": "1234567890",
  "componentGroupName": "ABC Insurance Policy",
  "componentGroupSequence": 1,
  "id": "1234567890",
  "name": "ABC Insurance Policy",
  "sequence": 1
}
```

Property Name	Type	Description	Filter Group and Version	Available Version
attributes	Attribute Detail[]	List of attributes.	Small, 63.0	63.0
componentGroupId	String	ID of the component group.	Small, 63.0	63.0
componentGroupName	String	Name of the component group.	Small, 63.0	63.0
componentGroupSequence	Integer	Sequence of the component group.	Small, 63.0	63.0
id	String	ID of the product.	Small, 63.0	63.0
name	String	Name of the product.	Small, 63.0	63.0
sequence	Integer	Sequence of the product.	Small, 63.0	63.0

Insurance Brokerage Standard Invocable Actions

Learn more about the standard invocable actions available with Insurance Brokerage.

[Compute Producer Splits Action](#)

Compute the producer splits for the producers associated with an Insurance Policy, for a Commission Statement Line Item.

[Create Producer Commissions Action](#)

Create records for the commissions that producers receive for the insurance policy associated with the specified commission statement line item, and update the commission statement line item record status.

[Find Insurance Policy Action](#)

Get the insurance policy associated with a commission statement line item that matches the specified criteria, and update the status of the commission statement line item record.

Compute Producer Splits Action

Compute the producer splits for the producers associated with an Insurance Policy, for a Commission Statement Line Item.

This action is available in API version 63.0 and later.

Special Access Rules

The Compute Producer Splits action is available in Enterprise, Unlimited, and Developer Editions where where Financial Service Cloud and Insurance Brokerage are enabled. To use this action, you need Insurance Commission Management and Insurance Brokerage User permission sets.

Supported REST HTTP Methods

URI

/services/data/v64.0/actions/standard/computeProducerSplits

Formats

JSON

HTTP Methods

GET, POST

Authentication

Authorization: Bearer ***token***

Inputs

Input	Details
commissionStatementLineItemId	Type String Description ID of the commission statement line item record.

Outputs

Output	Details
producerCommissions	Type sObject Description Collection of producer commission records that are computed.

Example

POST

This sample request is for the Compute Producer Splits action.

```
{
  "inputs": [
    {
      "commissionStatementLineItemId": "1Atxx0000004DeaCAE"
    }
  ]
}
```

This is the sample response for the Compute Producer Splits action.

```
[
  {
    "actionName": "computeProducerSplits",
    "errors": null,
    "invocationId": null,
    "isSuccess": true,
    "outcome": null,
    "outputValues": {
      "producerCommissions": [
        {
          "attributes": {
            "type": "ProducerCommission"
          },
          "CommissionStatementLineItemId": "1Atxx0000004C92CAE",
          "ProducerSplitAssignmentId": "1AIxx0000004C92GAE",
          "CommissionAmount": 4000,
          "InsurancePolicyId": "0YTxx0000000001GAA",
          "ProducerId": "0Yxxx0000000001CAA",
          "ProducerSplitArngLineItemId": "1BHxx0000004C92GAE",
          "Name": "CSLI-0000000001-Amol-Commission"
        }
      ]
    },
    "sortOrder": -1,
    "version": 1
  }
]
```

Create Producer Commissions Action

Create records for the commissions that producers receive for the insurance policy associated with the specified commission statement line item, and update the commission statement line item record status.

This action is available in API version 63.0 and later.

Special Access Rules

The Create Producer Commissions action is available in Enterprise, Unlimited, and Developer Editions where where Financial Service Cloud and Insurance Brokerage are enabled. To use this action, you need Insurance Commission Management and Insurance Brokerage User permission sets.

Supported REST HTTP Methods

URI

```
/services/data/v64.0/actions/standard/createProducerCommissions
```

Formats

JSON

HTTP Methods

GET, POST

Authentication

Authorization: Bearer *token*

Inputs

Input	Details
producerCommissions	Type sObject Description A collection of computed producer commission records to be created.
commissionStatementLineItemId	Type String Description ID of the commission statement line item record.
successStatus	Type String Description The status that the associated commission statement line item is set to when the producer commission record is created.
failureStatus	Type String Description The status that the associated commission statement line item is set to when the producer commission record isn't created.

Outputs

Output	Details
producerCommissionIds	Type String Description A collection of IDs of the producer commission records that are created.

Example

POST

This sample request is for the Create Producer Commissions action.

```
{
  "inputs": [
    {
      "commissionStatementLineItemId": "1Atxx0000004DeaCAE",
      "producerCommissions": [
        {
          "CommissionAmount": 3000,
          "InsurancePolicyId": "0YTxx00000002BVGAY",
          "ProducerId": "0Yxxx00000001IfCAI",
          "ProducerSplitArngLineItemId" : "1BHxx0000004CXEGA2",
          "CommissionStatementLineItemId": "1Atxx0000004DeaCAE",
          "Name": "Sanjeev"
        },
        {
          "InsurancePolicyId": "0YTxx00000002BVGAY",
          "ProducerId": "0Yxxx00000001KHCAI",
          "ProducerSplitArngLineItemId" : "1BHxx0000004CXEGA2",
          "CommissionStatementLineItemId": "1Atxx0000004DeaCAE",
          "Name": "Deepak"
        }
      ]
    }
  ]
}
```

This is the sample response for the Create Producer Commissions action.

```
[
  {
    "actionName": "createProducerCommissions",
    "errors": null,
    "invocationId": null,
    "isSuccess": true,
    "outcome": null,
    "outputValues": {
      "producerCommissionIds": [
        "0kAxx0000000001"
      ]
    },
    "sortOrder": -1,
    "version": 1
  }
]
```

Find Insurance Policy Action

Get the insurance policy associated with a commission statement line item that matches the specified criteria, and update the status of the commission statement line item record.

This action is available in API version 63.0 and later.

Special Access Rules

The Find Insurance Policy action is available in Enterprise, Unlimited, and Developer Editions where where Financial Service Cloud and Insurance Brokerage are enabled. To use this action, you need Insurance Commission Management and Insurance Brokerage User permission sets.

Supported REST HTTP Methods

URI

/services/data/v64.0/actions/standard/findInsurancePolicy

Formats

JSON

HTTP Methods

GET, POST

Authentication

Authorization: Bearer *token*

Inputs

Input	Details
commissionStatementLineItemId	Type String Description ID of the commission statement line item record.
matchingCriteriaPattern	Type String Description String that identifies the corresponding insurance policy.
failureStatus	Type String Description Indicates the status of the commission statement line item when the item's processing fails.

Outputs

Output	Details
commissionStatementLineItem	<p>Type</p> <p>sObject</p> <p>Description</p> <p>Commission statement line item record populated with the insurance policy ID.</p>

Example

POST

This sample request is for the Find Insurance Policy action.

```
{
  "inputs": [
    {
      "commissionStatementLineItemId": "1Atxx0000004DeaCAE",
      "matchingCriteriaPattern": "InsurancePolicyNumber:PolicyName AND
InsurancePolicyEffectiveDate:EffectiveFromDate AND LineOfCoverage:LineOfCoverage AND
ClientAccountName:NameInsured.Name"
    }
  ]
}
```

This is the sample response for the Find Insurance Policy action.

```
{
  "actionName": "findInsurancePolicy",
  "errors": null,
  "invocationId": null,
  "isSuccess": true,
  "outcome": null,
  "outputValues": {
    "commissionStatementLineItem": {
      "attributes": {
        "type": "CommissionStatementLineItem",
        "url":
"/services/data/v64.0/subjects/CommissionStatementLineItem/1Atxx0000004C92CAE"
      },
      "InsurancePolicyEffectiveDate": "2025-02-01",
      "LineOfCoverage": "Medical",
      "ReferenceRecordId": "0YTxx0000000001GAA",
      "InsurancePolicyNumber": "test",
      "Id": "1Atxx0000004C92CAE",
      "ClientAccountName": "test"
    }
  },
  "sortOrder": -1,
  "version": 1
}
```

Insurance Brokerage Metadata API Types

Metadata API enables you to access some types and feature settings that you can customize in the user interface.

[Flow for Insurance Brokerage](#)

Represents the metadata associated with a flow. Use Flow to create an application that takes users through a series of pages to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

Flow for Insurance Brokerage

Represents the metadata associated with a flow. Use Flow to create an application that takes users through a series of pages to query and update records in the database. You can also execute logic and provide branching capability based on user input to build dynamic applications.

FlowActionCall

Insurance Brokerage exposes additional `actionType` values for the `FlowActionCall` Metadata type.

Field Name	Field Type	Description
<code>actionType</code>	<code>InvokableActionType</code> (enumeration of type string)	<p>Required.</p> <p>The action type. Additional valid values only for Insurance Brokerage include:</p> <ul style="list-style-type: none">• <code>computeProducerSplits</code>—Compute the producer splits for the producers associated with an Insurance Policy, for a Commission Statement Line Item.• <code>createProducerCommissions</code>—Create records for the commissions that producers receive for the insurance policy associated with the specified commission statement line item, and update the commission statement line item record status.• <code>findInsurancePolicy</code>—Get the insurance policy associated with a commission statement line item that matches the specified criteria, and update the status of the commission statement line item record.

Insurance Brokerage Apex Reference

Apex is a strongly typed, object-oriented programming language that developers can use to execute flow and transaction control statements on the Salesforce Platform server, in conjunction with calls to the API. Insurance Brokerage Apex reference includes built-in Apex classes and interfaces, grouped by namespace.

[ConnectApi Namespace](#)

The `ConnectApi` namespace (also called Connect in Apex) in Insurance Brokerage provides classes for accessing the same data available in the Insurance Brokerage Business APIs.

ConnectApi Namespace

The `ConnectApi` namespace (also called Connect in Apex) in Insurance Brokerage provides classes for accessing the same data available in the Insurance Brokerage Business APIs.

For information about working with the `ConnectApi` classes, see [Connect in Apex](#).

[InsuranceBrokerageFamily Class](#)

Manage policy lifecycle, brokerage commissions, and plan benefits.

[ConnectApi Input Classes](#)

Some `ConnectApi` methods take arguments that are instances of `ConnectApi` input classes.

[ConnectApi Output Classes](#)

Most `ConnectApi` methods return instances of `ConnectApi` output classes.

InsuranceBrokerageFamily Class

Manage policy lifecycle, brokerage commissions, and plan benefits.

Namespace

`ConnectApi`

[InsuranceBrokerageFamily Methods](#)

These methods are for `InsuranceBrokerageFamily`. All methods are static.

InsuranceBrokerageFamily Methods

These methods are for `InsuranceBrokerageFamily`. All methods are static.

[bulkCancelInsurancePolicy\(insurancePolicyBulkInput\)](#)

Cancel policies in bulk for the specified Insurance Policy IDs.

[bulkRenewInsurancePolicy\(insurancePolicyBulkInput\)](#)

Renew policies in bulk for the specified Insurance Policy IDs.

[getCommissionProcessing\(commissionProcessingInput\)](#)

Process commission statements in an asynchronous manner by using the preconfigured batch job.

[renewInsurancePolicy\(insurancePolicyProcessInput, policyId\)](#)

Renew policy for the specified insurance policy ID.

`bulkCancelInsurancePolicy (insurancePolicyBulkInput)`

Cancel policies in bulk for the specified Insurance Policy IDs.

API Version

64.0

Requires Chatter

No

Signature

```
public static ConnectApi.InsurancePolicyBulkOutputRepresentation  
bulkCancelInsurancePolicy(ConnectApi.InsurancePolicyBulkInputRepresentation  
insurancePolicyBulkInput)
```

```
ConnectApi.InsuranceBrokerageFamily, bulkCancelInsurancePolicy,  
[ConnectApi.InsurancePolicyBulkInputRepresentation],  
ConnectApi.InsurancePolicyBulkOutputRepresentation
```

Parameters

insurancePolicyBulkInput

Type: [ConnectApi.InsurancePolicyBulkInputRepresentation](#) on page 181

Input representation of the request to cancel policies in bulk.

Return Value

Type: [ConnectApi.InsurancePolicyBulkOutputRepresentation](#) on page 183

Usage

Use this method to cancel insurance policies. The JSON input includes a field value map of insurance policy fields, such as CancellationEffectiveDate, Status, and CancellationReason, and a list of insurance policy IDs. After the request is successfully sent to the message queue, policies are canceled asynchronously, and a success or failure notification is sent to the user after the completion of the request.

bulkRenewInsurancePolicy (insurancePolicyBulkInput)

Renew policies in bulk for the specified Insurance Policy IDs.

API Version

64.0

Requires Chatter

No

Signature

```
public static ConnectApi.InsurancePolicyBulkOutputRepresentation
bulkRenewInsurancePolicy (ConnectApi.InsurancePolicyBulkRenewInputRepresentation
insurancePolicyBulkInput)
```

```
ConnectApi.InsuranceBrokerageFamily, bulkRenewInsurancePolicy,
[ConnectApi.InsurancePolicyBulkRenewInputRepresentation],
ConnectApi.InsurancePolicyBulkOutputRepresentation
```

Parameters

insurancePolicyBulkInput

Type: [ConnectApi.InsurancePolicyBulkRenewInputRepresentation](#) on page 181

Input representation of the request to renew insurance policies in bulk.

Return Value

Type: [ConnectApi.InsurancePolicyBulkOutputRepresentation](#) on page 183

Usage

Insurance Policy Bulk Renewal API takes a JSON input that contains a list of insurance policy IDs that must be renewed along with the `fieldValueMap` and `sameCarrier` details. The API submits the request to a queue, creates policies, along with child records by using the context definition selected in setup. The service also overrides the `fieldValueMap` in the new policy for each policy ID passed in the request. Users receive success and failure notifications on completion.

Keep these considerations in mind when you use this method.

- The `RenewedFromPolicyId` value of the new policy is set to the current policy ID and `IsRenewedPolicy` field of the new policy is set to `true`.
- The name of the new policy is a copy of `{ExistingName}`.
- If the policy is renewed with the same carrier, the `PriorPolicyId` field of the new policy is updated with the current policy ID.
- The `fieldValueMap` expects the keys to be API name and is case-sensitive.

getCommissionProcessing(commissionProcessingInput)

Process commission statements in an asynchronous manner by using the preconfigured batch job.

API Version

63.0

Requires Chatter

No

Signature

```
public static ConnectApi.InsuranceCommissionProcessingOutputRepresentation  
getCommissionProcessing(ConnectApi.InsuranceCommissionProcessingInputRepresentation  
commissionProcessingInput)
```

```
ConnectApi.InsuranceBrokerageFamily, getCommissionProcessing,  
[ConnectApi.InsuranceCommissionProcessingInputRepresentation],  
ConnectApi.InsuranceCommissionProcessingOutputRepresentation
```

Parameters

commissionProcessingInput

Type: [ConnectApi.InsuranceCommissionProcessingInputRepresentation](#) on page 181

Input representation details of the insurance commission processing API.

Return Value

Type: [ConnectApi.InsuranceCommissionProcessingOutputRepresentation](#) on page 182

Usage

Commission Statement Processing API uses a JSON input that contains a list of `commissionStatementIds`. The service processes the commission statement asynchronously by using a batch job. The batch job calls the processing flow to process each commission statement line item. When the processing is complete, the service updates the commission statement and sends a notification to the user.

renewInsurancePolicy(insurancePolicyProcessInput, policyId)

Renew policy for the specified insurance policy ID.

API Version

63.0

Requires Chatter

No

Signature

```
public static ConnectApi.InsurancePolicyProcessOutputRepresentation  
renewInsurancePolicy(ConnectApi.InsurancePolicyRenewInputRepresentation  
insurancePolicyProcessInput, String policyId)
```

```
ConnectApi.InsuranceBrokerageFamily, renewInsurancePolicy,  
[ConnectApi.InsurancePolicyRenewInputRepresentation, String],  
ConnectApi.InsurancePolicyProcessOutputRepresentation
```

Parameters

*insurancePolicyProcessInput*Type: [ConnectApi.InsurancePolicyRenewInputRepresentation](#) on page 182

Input representation details of the request to renew an insurance policy.

policyId

Type: String

ID of the insurance policy that's to be renewed.

Return Value

Type: [ConnectApi.InsurancePolicyProcessOutputRepresentation](#) on page 183

Usage

The Insurance Policy Renewal API takes a JSON input that contains the insurance policy ID that must be renewed along with the `fieldValueMap` and `sameCarrier` details. The API submits the request to a queue, creates a policy, along with child records by using the context definition selected in setup. The service also overrides the `fieldValueMap` in the new policy for the policy ID passed in the request. Users receive success and failure notifications on request completion.

Keep these considerations in mind when you use this method.

- The `RenewedFromPolicyId` value of the new policy is set to the current policy ID and `IsRenewedPolicy` field of the new policy is set to `true`.
- If the policy is renewed with the same carrier, the `PriorPolicyId` field of the new policy is updated with the current policy ID.
- The `fieldValueMap` expects the keys to be API name and is case-sensitive.

ConnectApi Input Classes

Some `ConnectApi` methods take arguments that are instances of `ConnectApi` input classes.

Input classes are concrete unless marked abstract in this documentation. Concrete input classes have public constructors that have no parameters.

Some methods have parameters that are typed with an abstract class. You must pass in an instance of a concrete child class for these parameters.

Most input class properties can be set. Read-only properties are noted in this documentation.

Insurance Brokerage includes these Apex input classes.

[ConnectApi.InsuranceCommissionProcessingInputRepresentation](#)

Input representation details of the insurance commission processing resource. The input includes the `commissionStatementIds` property required to process commission data.

[ConnectApi.InsurancePolicyBulkInputRepresentation](#)

Input representation details of the cancel policies in bulk resource. The input includes the list of insurance policy IDs to be canceled and the details of the associated fields of the insurance policies.

[ConnectApi.InsurancePolicyBulkRenewInputRepresentation](#)

Input representation details of the renew insurance policies in bulk resource. The input includes the list of insurance policy IDs to be renewed and the details of the associated fields of the insurance policies.

[ConnectApi.InsurancePolicyRenewInputRepresentation](#)

Input representation details of the resource to renew an insurance policy. The policy can be renewed from the same insurance carrier or a different carrier based on the `sameCarrier` value passed in the request.

ConnectApi.InsuranceCommissionProcessingInputRepresentation

Input representation details of the insurance commission processing resource. The input includes the `commissionStatementIds` property required to process commission data.

Property	Type	Description	Required or Optional	Available Version
<code>commissionStatementIds</code>	Object	List of commission statement IDs to be processed.	Required	63.0

ConnectApi.InsurancePolicyBulkInputRepresentation

Input representation details of the cancel policies in bulk resource. The input includes the list of insurance policy IDs to be canceled and the details of the associated fields of the insurance policies.

Property	Type	Description	Required or Optional	Available Version
<code>fieldValueMap</code>	Object	JSON object that contains the name-value pairs that store the details of the insurance policy.	Required	64.0
<code>insurancePolicyIds</code>	Object	List of Insurance policy IDs.	Required	64.0

ConnectApi.InsurancePolicyBulkRenewInputRepresentation

Input representation details of the renew insurance policies in bulk resource. The input includes the list of insurance policy IDs to be renewed and the details of the associated fields of the insurance policies.

Property	Type	Description	Required or Optional	Available Version
<code>fieldValueMap</code>	Object	JSON object that contains the name-value pairs that store the details of the insurance policy.	Required	64.0
<code>insurancePolicyIds</code>	Object	List of insurance policy IDs.	Required	64.0
<code>sameCarrier</code>	Boolean	Indicates whether the insurance policy is renewed with the same carrier (<code>true</code>) or not (<code>false</code>). If a policy is renewed with the same carrier, the <code>PriorPolicyId</code> field in the new policy is updated with the current policy ID. If not, the <code>PriorPolicyId</code> field is left blank.	Required	64.0

ConnectApi.InsurancePolicyRenewInputRepresentation

Input representation details of the resource to renew an insurance policy. The policy can be renewed from the same insurance carrier or a different carrier based on the `sameCarrier` value passed in the request.

Property	Type	Description	Required or Optional	Available Version
<code>fieldValueMap</code>	<code>Object</code>	JSON object that contains name-value pairs that store the details of the insurance policy.	Required	63.0
<code>sameCarrier</code>	<code>Boolean</code>	Indicates whether the insurance policy is renewed with the same carrier (<code>true</code>) or not (<code>false</code>). If a policy is renewed with the same carrier, the <code>PriorPolicyId</code> field in the new policy is updated with the current policy ID. If not, the <code>PriorPolicyId</code> field is left blank.	Required	63.0

ConnectApi Output Classes

Most `ConnectApi` methods return instances of `ConnectApi` output classes.

All properties are read-only, except for instances of output classes created within the test code.

All output classes are concrete unless marked abstract in this documentation.

All concrete output classes have no-argument constructors that you can invoke only from the test code. See [Testing ConnectApi Code](#).

Insurance Brokerage includes these Apex output classes.

[ConnectApi.InsuranceCommissionProcessingOutputRepresentation](#)

Output representation details of the commission processing resource.

[ConnectApi.InsuranceOutputRepresentation](#)

Output representation details of the bulk renew policy and bulk cancel policy requests.

[ConnectApi.InsurancePolicyBulkOutputRepresentation](#)

Output representation details of the bulk cancel policy resource.

[ConnectApi.InsurancePolicyProcessOutputRepresentation](#)

Output representation details of the insurance policy renewal resource.

ConnectApi.InsuranceCommissionProcessingOutputRepresentation

Output representation details of the commission processing resource.

Property Name	Type	Description	Available Version
<code>message</code>	<code>String</code>	Status message of the API execution.	63.0
<code>status</code>	<code>String</code>	Status of the API execution. Valid values are: <ul style="list-style-type: none"> • <code>Success</code> • <code>Fail</code> 	63.0

ConnectApi.InsuranceOutputRepresentation

Output representation details of the bulk renew policy and bulk cancel policy requests.

This class is abstract.

Superclass of [ConnectApi.InsurancePolicyBulkOutputRepresentation](#) .

Property Name	Type	Description	Available Version
errors	ConnectApi.InsuranceOutputRepresentation List of errors	List of the errors encountered during the processing of the API request.	63.0
message	String	Status message of the API execution.	63.0
status	String	Status of the API execution. Valid values are: <ul style="list-style-type: none"> • Success • Fail 	63.0

ConnectApi.InsurancePolicyBulkOutputRepresentation

Output representation details of the bulk cancel policy resource.

Subclass of [ConnectApi.InsuranceOutputRepresentation](#).

No additional properties.

ConnectApi.InsurancePolicyProcessOutputRepresentation

Output representation details of the insurance policy renewal resource.

Property Name	Type	Description	Available Version
policyId	String	ID of the insurance policy to be returned.	63.0