



Salesforce Console Implementation Guide for Salesforce Classic

Salesforce, Spring '17



CONTENTS

INTRODUCTION	1
Salesforce Console	1
Who Should Read This Guide	1
PLAN AND REVIEW LIMITATIONS	2
Plan Your Implementation	2
Planning Questions	2
Implementation Tips	3
Limitations	4
Configurable Features	6
SET UP A CONSOLE	7
Set Up a Service Console	7
Get Started with Launch Pad Templates	7
Set Up a Sales Console	8
Set Up a Custom Console	9
Brand a Console	10
Add a Highlights Panel	11
Configure a Highlights Panel	11
Choose How Lists Display	12
Optimize with Responsive Lists	12
Whitelist Domains	13
Configure Push Notifications	14
Customize Keyboard Shortcuts	15
Set Up Interaction Logs	16
Turn On Interaction Logs	16
Assign Interaction Logs	17
Assign a License to Service Users	17
Assign a Permission Set to Sales Users	18
Hide the Console in the Lightning Experience App Launcher	18
LEARN ABOUT MORE CONSOLE FEATURES	19
Multi-Monitor Components	19
Save User Sessions	19
History Component	20
Call Center Softphones	20
Knowledge	21
Live Agent	21
Macros	22

SUPER CHARGE YOUR CONSOLE	23
Customize with Custom Components	23
Component Process Overview	23
Component Implementation Tips	24
Add Components to Page Layouts	25
Add Quick Actions to the Sidebar	26
Sidebar Styles for Components	27
Add Components to Apps	28
Assign a Component to an App	28
More Resources	28
INDEX	30

INTRODUCTION

Salesforce Console

A Salesforce Console app is a tab-based workspace that lets you manage multiple records on a single screen. The console inherently minimizes clicking and scrolling so that you can quickly find, update, and create records. Thousands of customers rely on the Salesforce Console to boost productivity in fast-paced service and sales environments.

The console is perfect for Sales and Service folks. Support agents can use multiple applications at once and keep the context of cases as priorities change. Sales reps have all the tools at their fingertips to cultivate leads, close deals, and manage relationships.

Administrators can customize a console to determine everything from the objects in the navigation tab, to whether you can place phone calls with a softphone. Customizations can also include displaying knowledge base articles alongside cases, creating custom actions with keyboard shortcuts, or chatting with customers over the web with Live Agent.

Developers can help solve unique business problems with custom console components and use the Salesforce Console Integration Toolkit API to add third-party integrations.



Note: You can create console apps in Salesforce Classic and Lightning Experience.

This release contains a beta version of Lightning console apps, which means it's a high-quality feature with known limitations. Lightning console apps aren't generally available unless or until Salesforce announces its general availability in documentation or in press releases or public statements. We can't guarantee general availability within any particular time frame or at all. Make your purchase decisions only on the basis of generally available products and features.

When working with Lightning console apps, keep the following things in mind.

- Lightning console apps don't yet have full parity with Salesforce Classic console apps. For example, some features in Salesforce Classic console apps, such as keyboard shortcuts and macros, aren't available in Lightning console apps.
- You can't migrate Salesforce Classic console apps to Lightning Experience.

EDITIONS

Available in Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited, and Developer** Editions with Sales Cloud and Service Cloud

Who Should Read This Guide

The Salesforce Console Implementation Guide is for administrators who want to plan and implement a one time, basic setup of a console in Salesforce Classic.



Important: This guide covers the setup and customization of Salesforce Classic console apps only. For information about Lightning Experience console apps, see [Salesforce Console in Lightning Experience \(Beta\)](#) in the Salesforce online help.


Administrators create consoles for sales or service teams who quickly access multiple records to help customers. Optionally, developers can customize, extend, and integrate a console with other systems using console components and an API known as the Salesforce Console Integration Toolkit. For more information on customizing a console with code, see [Customize a Console with Custom Components in Salesforce Classic](#).

PLAN AND REVIEW LIMITATIONS

Plan Your Salesforce Console Implementation

Before you set up a console, determine whether it's for Service, Sales, or a unique business need. Think about who will use it and what information your console users need most.

You can set up multiple consoles in your Salesforce org so long as you don't exceed your edition's custom apps limit. If you're already using a console for Service, you can create a console for Sales without purchasing separate licenses.

 **Note:** Lightning console apps don't yet have full parity with Salesforce Classic console apps. For example, some features in Salesforce Classic console apps, such as keyboard shortcuts and macros, aren't available in Lightning console apps.

You can't migrate Salesforce Classic console apps to Lightning Experience.

Salesforce Classic Consoles

If you're assigned the Service Cloud User feature license, you can review and customize an out-of-the-box console by selecting **Sample Console** from the app menu.

Lightning Experience Consoles

Salesforce provides standard Lightning console apps for Sales and Service. To access these apps, you must be assigned the correct feature license. Go to the App Launcher and click **Service Console** or **Sales Console**. These apps are designed to meet all your Service and Sales needs and can be customized.

EDITIONS

Available in Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited, and Developer** Editions with Sales Cloud and Service Cloud

Salesforce Console Planning Questions

Review these questions to help you get the most out of your console implementation.

Who will use a console in your Salesforce org?

A console is designed for users who want to find, update, and create records quickly. If users' profiles are assigned to a console app, they can access it. Before creating a console, we recommend determining which of your existing user profiles will benefit the most from console access, such as profiles assigned to support agents or a telesales team. You can also consider creating a profile or cloning an existing profile from which you can assign users to a console.

Have you purchased enough feature licenses?

Salesforce users must have a feature license to use a console. After you set up a console, you can manually assign the licenses to users.

Who will implement console apps for your company?

A console is an app, consisting of a group of tabs working as a unit to provide application functionality. Salesforce provides standard apps, such as Sales and Service. You create a console as a custom app that users select from the app menu and App Launcher. If there is a user in your company who is familiar with creating apps in Salesforce, have that person implement console apps.

Which objects do you want users to access in the console?

The navigation tab in Salesforce Classic and the navigation bar in Lightning Experience, lets users select objects, such as cases, accounts, or contacts, and view that object's home page. We recommend that you add objects that function as the center of your business processes. For example, the center of support agents' business process is most likely cases, so make cases available to agents in the console.

How do you want records to display in a console when they're selected?

A console displays all records as tabs so that users can quickly find, update, and create records on one screen. Primary tabs display the main item to work on, such as an account. Subtabs display related records, such as cases or contacts on an account.

Salesforce Classic

You can select which records display as primary tabs and subtabs. Pick the center of your business process as a primary tab, and related items as subtabs. For example, you can set accounts as a primary tab, and cases and contacts as subtabs. You can also choose how records display when they're selected outside of primary tabs or subtabs, for example, when users select cases from list views or search results.

Lightning Experience

In Lightning console apps, primary tabs are called workspace tabs. You can't specify which records display as workspace tabs and subtabs. Records are opened as workspace tabs when selected from the list view associated with the current navigation bar item. Records opened from a workspace tab display as subtabs. For example, if you select accounts from the navigation bar and then click an account in the list, the account is opened as a workspace tab. Then if you click a contact name from the account page, the contact record is opened as a subtab.

What features can I use in the console?**Salesforce Classic**

Available features include things like Highlight Panels, Interaction Logs, Live Agent, a Knowledge sidebar, whitelisting domains, choosing how lists display, setting up push notifications, keyboard shortcuts, and more.

For a list of features that you can implement in your console, see [Console Configurable Features in Salesforce Classic](#).

Lightning Experience

Customizing a Lightning console app is different than a Salesforce Classic console. For more information, see [Customize Your Lightning Console App with Utilities](#) and [Customize Your Lightning Console App with Lightning Pages](#).

Do you want users to receive and make phone calls from a console?

You can integrate Salesforce with third-party computer-telephony integration systems (CTI) using Open CTI. For more information, see [Salesforce Call Center](#).

Do you want to customize, extend, or integrate a console with other systems?

Create components to customize, extend, or integrate the Salesforce console with other systems. For example, you might want to create components that display third-party apps or content when users view specific pages.

What is the Agent Console?

The Salesforce console improves upon the Agent console by providing you with more options and more advanced technologies.

Salesforce Console Implementation Tips

Review these tips to make sure that you're getting the most out of your console implementation.

General Implementation Tips

- A console respects all the security and sharing settings in your Salesforce org.
- For best performance, we recommend that console users adopt the Google Chrome™ browser and machines with 8 GB of RAM. Keep in mind that 32-bit systems are subject to memory limitations. To take full advantage of the added RAM, use 64-bit versions of both the OS and the browser.
- In Salesforce Classic consoles, you can further improve performance by enabling Tab Limits. To enable this feature, from Setup, go to the Apps page and edit your console.
- To see any setup changes to a console, refresh your browser.

EDITIONS

Available in Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited, and Developer** Editions with Sales Cloud and Service Cloud


Salesforce Console for Service Implementation Tips

- To use a console for Service, users must be assigned the Service Cloud User feature license.
- To use Live Agent in a console, users must be assigned the Live Agent User feature license.
- If Salesforce Knowledge is set up, turn on the Knowledge sidebar so that console users can automatically see relevant articles alongside any cases that they're working on.

Salesforce Classic Console Limitations

Learn about display, access, and customization limitations for a console.

Display Limitations


- A Salesforce console is available only in Supported Browsers for Salesforce Classic. If you use an unsupported browser, you get a degraded experience.
 -  **Tip:** For best performance, we recommend that console users adopt the Google Chrome™ browser and machines with 8 GB of RAM. Keep in mind that 32-bit systems are subject to memory limitations. To take full advantage of the added RAM, use 64-bit versions of both the OS and the browser.
- A console offers limited support for right-to-left languages, such as Arabic or Hebrew. Since right-to-left languages aren't fully supported, some features don't appear as expected in a console, such as Case Feed, Live Agent, and Salesforce Knowledge.
- All Salesforce CRM Content items display as primary tabs in a console.
- When using Social Accounts, Contacts, and Leads, the Social Viewer window doesn't align properly within a console.
- Record home pages, including their standard Tools sections, don't display in a console.

EDITIONS

Available in Salesforce Classic

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

Behavior and Access Limitations

- Back buttons on browsers are not supported for a console and don't work as expected.
- A Salesforce console doesn't meet accessibility requirements.
- Visualforce pages don't refresh when you click  and select **Refresh all primary tabs** or **Refresh all subtabs**. In addition, if you override a detail page with a Visualforce page that uses related lists, any new records created using that related list aren't displayed until the page is reopened in a new tab.
- If you edit a page layout in the console, such as adding a primary tab component like an interaction log, the updated layout isn't displayed until the page is closed and then reopened in a new primary tab.
- You can't directly edit the fields on the highlights panel; however, when you change those fields on a record, the highlights panel is automatically updated.
- Some objects haven't been fully adapted to a console, such as reports, dashboards, and Chatter. These and other objects are marked with an asterisk (*) in the console setup area and might cause unexpected behaviors. For example: (1) When viewing opportunity splits, clicking **Cancel** doesn't close the subtab; instead, the page detail information is opened in a subtab; (2) opportunity line items and quote line items don't automatically refresh when changed—you might need to refresh the page to see the latest changes.
- If you leave a console and return to it, any subtabs that were displayed might not automatically reload. Also, if you leave a console by selecting an app from the Force.com app menu, your browser's **Back** and **Forward** buttons might not work as expected.
- If you save changes in a lookup component, and you have unsaved changes in another lookup component, the unsaved changes are lost without a warning message.

- If you click a custom link or a link in a text field, such as **http://www.salesforce.com/** in a case `Description`, the link appears in a popup window instead of a tab.
- If you refresh the Change Case Status page in a console, an `Unable to Access Page` error appears.
- If Chatter is disabled, you can't use pinning in search results.

Customization Limitations

To edit a Salesforce Classic console app, you must be in Salesforce Classic. From Setup, enter `Apps` in the `Quick Find` box, then select **Apps** and click **Edit** next to your console app.


- The console doesn't support Chatter Messenger.
- Some Visualforce pages don't display correctly in a console unless they're updated to show in it.
- The console doesn't support Visualforce pages that use the **Require CSRF protection on GET requests** preference. If you use Visualforce pages with CSRF protection, your agents can receive errors when trying to access the page.
- If you enable clickjack protection for Visualforce pages in your Salesforce org's security settings, Visualforce pages won't work correctly.
- Call center users can only use a softphone in a Salesforce console if they're using a CTI adapter built with [Open CTI](#).
- Live Agent is only available for a console with the Service Cloud.
- The highlights panel for Live Agent Transcripts is editable, but you can't turn it on for users.
- You can't add a console to customer portals or partner portals.

External Object Limitations

- External objects are available in the Salesforce Console in Salesforce Classic only. Other consoles aren't supported.
- External objects haven't been fully adapted to a console and can cause unexpected behaviors. Other objects also haven't been fully adapted to a console, but those objects are marked with an asterisk (*) in the console setup area. External objects aren't marked with asterisks.
- The external objects' associated external data sources must have the **High Data Volume** option deselected. This requirement doesn't apply to the cross-org adapter for Salesforce Connect.

Console Configurable Features in Salesforce Classic

An administrator must turn on and configure most console features to determine how you view and work with data.

 **Note:** To edit a Salesforce Classic console app, you must be in Salesforce Classic. From Setup, enter *Apps* in the *Quick Find* box, then select **Apps** and click **Edit** next to your console app.

Feature	Automatically Enabled?	Available in a Console for Sales	Available in a Console for Service
Highlights Panel		✓	✓
Brand a Console		✓	✓
Pinned Lists	✓	✓	✓
Save User Sessions	✓	✓	✓
Bookmarks and Recent Tabs	✓	✓	✓
Keyboard Shortcuts	✓	✓	✓
Multi-Monitor Components	✓	✓	✓
Console Components		✓	✓
Whitelist Domains		✓	✓
Push Notifications		✓	✓
Responsive Lists		✓	✓
Interaction Logs		✓	✓
Softphone Integration		✓	✓
Knowledge Sidebar		✓	✓
Knowledge One Footer		✓	✓
Live Agent			✓

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

Configure a Salesforce Console for Service:

- "Customize Application"

Salesforce Console for Sales:

- "Customize Application"

AND


"Sales Console User"

SET UP A CONSOLE


Set Up a Service Console in Salesforce Classic

Create and customize a console app, then assign the Service Cloud User feature license to users.

1. [Customize highlight panels for all objects](#) and [turn them on](#).
2. [Create a console app and assign it to user profiles](#).
3. Optionally, for support agents:
 - [Brand your console](#).
 - [Choose how pinned lists display](#).
 - [Whitelist domains](#).
 - [Customize push notifications](#).
 - [Customize keyboard shortcuts](#).
 - [Turn on the Knowledge Sidebar](#).
 - [Set up interaction logs](#) and [assign them to user profiles](#).
 - Turn on Live Agent (available only for a console for service).

 **Note:** Set up your Live Agent implementation before adding it to your console. If you're adding Live Agent to an existing console, test your console before deploying it to your agents. Make sure your sidebar components look as expected.


4. [Assign users the Service Cloud User feature license](#).

 **Tip:** To add third-party integrations, or expand console customizations to new levels, advanced Salesforce admins and developers can create console components, or use code with an API named the Salesforce Console Integration Toolkit.

Set Up Service Cloud Features with Launch Pad Templates

Deploy and test Service Cloud features with just a few clicks. This suite of templates helps you set up some of the most popular Service Cloud features, such as time-saving console configurations and reports and dashboards essential for service organization.

To use templates, from the Setup **Quick Search** box, enter **Launch Pad**.

 **Tip:** It's best to try Launch Pad in your sandbox environment, so that you can test out new features without affecting your live production environment.

Launch Pad templates aren't supported for IE version 9 or earlier.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To set up a Salesforce console for Service:

- "Customize Application"
- AND
- "Service Cloud User"

EDITIONS

Available in Salesforce Classic

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

Set Up Sales Console in Salesforce Classic

To enable a console for the first time, contact Salesforce then assign users the correct permission set license and user permissions.

1. To have the console for sales enabled for your Salesforce org, contact Salesforce.
2. [Customize highlight panels for all objects](#) and [turn them on](#).
3. By default, your org has the “Sales Console User” permission set license. Assign this permission set license to users. The steps are the same as assigning any permission set license.
4. [Assign the Salesforce Console User Permission Set to Sales Console Users](#).
5. Optionally, for sales reps:
 - [Brand your console](#).
 - [Choose how pinned lists display](#).
 - [Whitelist domains](#).
 - [Customize push notifications](#).
 - [Customize keyboard shortcuts](#).
 - [Turn on the Knowledge Sidebar](#).
 - [Set up interaction logs](#) and [assign them to user profiles](#).
6. [Create a console app and assign it to user profiles](#).



Tip: To add third-party integrations, or expand console customizations to new levels, advanced Salesforce admins and developers can create console components, or use code with an API named the Salesforce Console Integration Toolkit.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To assign permission set licenses:

- “Manage Internal Users”

To create a Salesforce Console for Sales:

- “Customize Application”

AND

“Sales Console”

Set Up a Salesforce Console App in Salesforce Classic

To create a console, administrators must create a console app.



[Walk Through It](#) (Requires the “Sales Console” user permission or the Service Cloud User license)

Watch a Demo: [Plan Your Salesforce Console Implementation](#)

Each console is a custom app that users select from the Force.com app menu. The number of console apps you can create is based on your Salesforce org’s custom app limits.

1. From Setup, enter `Apps` in the `Quick Find` box, then select **Apps**, then click **Next**.
2. Click **New**, select `Console`, and click **Next**. Optionally, to copy an existing console, click **Clone** on the console’s detail page. Save a console that you’ve copied before you can update its cloned push notifications or keyboard shortcuts.
3. Type a label for the app, then click **Next**. The label can have a maximum of 40 characters, including spaces. This label is the app’s name in the Force.com app menu.
4. Optionally, add a custom logo. Click **Insert an image**, choose a file image from the document library, then click **Next**.
5. Select the items to include in the navigation tab, then click **Next**. You can’t add the following to the navigation tab: forecasts, ideas, answers, connections, portals, console, activities (tasks and events), the App Launcher, or buttons for customizing your org. Tab visibility settings, and the `Overwrite users' personal tab customizations` setting, don’t apply to the navigation tab. For example, you can access Accounts from the navigation tab even if `Accounts` are Tab Hidden on user profiles.
6. Choose how items that aren’t selected from a primary tab or subtab display, then click **Next**. For example, choose whether cases display as primary tabs or as subtabs on parent records when selected from list views, search results, or screen pops. Click:
 - `As a primary tab` to display the object’s records as primary tabs.
 - `As a subtab of` to display the object’s records as subtabs. From the drop-down list, choose the parent record where the subtabs appear. For example, if you want cases to appear as subtabs on contacts, select Contact Name (Contact) in the Cases row. If parent records don’t apply, subtabs appear as primary tabs. For example, if you make campaigns a subtab of leads but a lead doesn’t belong to a campaign, that campaign appears as a primary tab.
7. Optionally, if your org has Service Cloud and [Live Agent](#), select `Include Live Agent in this App`, and click **Next**. If you choose to include a chat workspace for Live Agent in your console, you can:
 - Choose records or pages to open as subtabs of chat sessions.
 - Select `Include Suggested Articles from Salesforce Knowledge in Live Agent` to display the articles tool in the chat workspace.
8. Assign the Salesforce console to user profiles:
 - a. Check the `Visible` box to choose the user profiles for which the app is available.
 - b. Check the `Default` box to set the app as that profile’s default app, meaning that new users with the profile see this app the first time they log in. Profiles with limits are excluded from this list.
9. Click **Save**.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To view apps:

- “View Setup and Configuration”

To manage apps:

- “Customize Application”
- AND
- “Service Cloud User” OR “Sales Console”


 **Tip:** To see any setup changes to a console, refresh your browser.

Brand a Salesforce Console in Salesforce Classic

You can brand a console by adding a custom logo to its header and specifying colors for its header, footer, and primary tabs.

1. From Setup, enter `Apps` in the `Quick Find` box, then select **Apps**.
2. Click `Edit` next to a console.
3. To add a custom logo, click **Insert an image**, and choose an image from the document library. Consider these requirements when choosing a custom app logo for a Classic app from the document library:
 - The image must be in GIF or JPEG format and less than 20 KB.
 - If the image is larger than 300 pixels wide by 55 pixels high, then it is scaled to fit.
 - For the best on-screen display, we recommend that you use an image with a transparent background.
 - The `Externally Available` checkbox must be selected on the document's properties so that users can view the image.
4. To specify a color, type a hex code in `Header Color`, `Footer Color`, or `Primary Tab Color`. For example, type `#0000FF` to define the header as blue.
5. Click **Save**.

To verify that the correct colors display, view your console.

 **Note:** Defining header or footer colors is not yet available for the Metadata API.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To define colors in the Salesforce console:

- "Customize Application"

Add a Highlights Panel to the Console in Salesforce Classic

The highlights panel is a customizable table of up to four columns at the top of every primary tab in a console. It helps users see key information at a glance.

A highlights panel can include any standard or custom fields that have already been added to a page layout. Each panel can have a minimum of one column of fields and a maximum of four columns. Each column can have one or two fields in top or bottom positions.

When users see a highlights panel, top fields appear in bold and in a larger font than bottom fields. Users can only see fields that they have permission to view—otherwise fields appear blank. When both fields appear blank in a column, the column appears blank.

To display highlights panels in a console, you must turn them on.

1. Edit any page layout and click **Layout Properties**.
2. Check `Highlights Panel`, and click **OK**.
3. Click **Save**

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To customize page layouts:


- "Customize Application"
- AND
- "Service Cloud User" OR
"Sales Console"


To view page layouts:

- "View Setup and Configuration"


Configure Highlights Panels for a Console in Salesforce Classic

The highlights panel is a customizable table of up to four columns at the top of every primary tab in a console. It helps console users see key information at a glance. To configure a highlights panel, edit any page layout.

To edit a highlights panel, hover over the panel on any page layout and click  or double-click the panel. To add or change fields, click the appropriate column and select fields from the `Top Field` and `Bottom Field` lists. If the fields you want to add aren't listed, you must first add them to the detail page layout. Fields can appear once on each highlights panel, and fields already in use appear as gray in the field selection list. Highlights panels don't support rich text area (RTA) fields, custom long-text area fields, or the Transcript object.

To remove fields, click the appropriate column and select `-None-` from the field lists. To add columns, click **Add Column**. To remove columns, hover over the column and click .

To save highlights panel changes, click **OK** in the Highlights Panel Properties dialog box. When you've finished editing the page layout, click **Save**. To display highlights panels in a console, you must turn them on.

 **Important:** If you leave a page layout before clicking **Save**, you'll lose your highlights panel changes.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To customize page layouts:

- "Customize Application"
- AND
- "Service Cloud User" OR
"Sales Console"

To view page layouts:

- "View Setup and Configuration"

Choose How Lists Display in a Salesforce Console in Salesforce Classic

Administrators can change how lists appear to console users.

In a console, you can display lists in one of three ways:

List Type	Description
Full screen, unpinned	Lists are visible only when the navigation tab is selected. When you select a record from the list, it opens in a new tab.
Pinned to top	Lists appear at the top of the page and are always visible. When you select a record from the list, it opens in a new tab below the list.
Pinned to left	Lists appear on the left side of the page and are always visible. When you select a record from the list, it opens in a new tab to the right of the list.

To change how lists display:

1. From Setup, enter *Apps* in the *Quick Find* box, then select **Apps**.
2. Select a console app.
3. Click **Edit**.
4. Choose how you want lists to display. If you choose pinned lists, specify a default size, in pixels or percentage of screen space.
5. Click **Save**.

Optimize Salesforce Console List Views with Responsive Lists in Salesforce Classic

Responsive lists resize dynamically and give your list views a modern look and feel. And optional row hovers let you preview a record without opening it.

Here's how to enable responsive lists:

1. From Setup, enter *Apps* in the *Quick Find* box, then select **Apps**.
2. Find the name of the console you want to modify, then click **Edit**.
3. Select **Enable Responsive Lists**.
4. Click **Save**.

The fields that appear in the hover overlay mirror the fields in the active list view. To customize which fields appear, edit the list view or create a custom list view. Keep in mind that the list view must be expanded wide enough for the Edit, Delete, and Create New View links to be visible.

Some more detail about list hovers:

- For cases, hovers appear when you mouse over the *Subject* field. For all other records, hovers appear when you over the *Name* field.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To choose how lists display:


- "Customize Application"










EDITIONS

Available in Salesforce Classic

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

- If the `Subject` or `Name` field isn't in the list view, the hover doesn't appear.
- You always see the `Description` field on case hovers, even though it can't be added to the list view.
- Responsive lists aren't supported for IE version 10 or earlier.

 **Tip:** If you want to keep the updated look and feel of responsive lists, but turn off the hover feature, edit your console and deselect **Enable List View Hovers**.

ACTION	CASE NUMBER ↑	CONTACT NAME	SUBJECT	STATUS	PRIORITY
  	00001000	Amos, Jon	Sample Case: Our Widgets have not been delivered.	Escalated	High
  	00001001	Stamos, Edward	Sample Case: Our Widgets have not been delivered.		
  	00001002	Stamos, Edward	S		

Sample Case: Our Widgets have not been delivered.

Case Number	Contact Name	Status
00001000	Amos, Jon	Escalated
Priority	Date/Time Opened	Case Owner Alias
High	7/11/2016 3:06 PM	AUser
Description		
The widget the customer requested weren't delivered on time.		

Whitelist Domains for a Salesforce Console in Salesforce Classic

Salesforce admins can let console users access domains outside of Salesforce. For example, you can add `www.example.com` to a console's whitelist so that console users can access that domain.

1. From Setup, enter `Apps` in the `Quick Find` box, then select **Apps**.
2. Select a console app.
3. Click **Edit**.
4. In `Whitelist Domains`, type the domains you want users to access, and separate multiple domains by commas. Don't include `http://` or `https://` because those prefixes are part of a URL, not a domain.

If your CTI phone is running on a server with a non-standard port, make sure to include the port number in your domain. For example, if your server is called `myserver` and your port number is `8500`, include `myserver:8500` in your whitelist.

5. Click **Save**.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To whitelist domains for a Salesforce console:

- "Customize Application"

Configure Push Notifications for a Salesforce Console in Salesforce Classic

Administrators can set up push notifications in a console so that users can see when a record they're working on has been changed by others.

Push notifications are visual indicators on lists and detail pages in a console that show when a record or field has changed during a user's session. For example, if two support agents are working on the same case, and one agent changes the `PRIORITY`, a push notification appears to the other agent so he or she spots the change and doesn't duplicate the effort.

Choose when push notifications appear and which objects and fields trigger them:

1. From Setup, enter `Apps` in the `Quick Find` box, then select **Apps**.
2. Select a console app, and click **Edit**.
3. In `Choose How Lists Refresh`, select when push notifications appear.

 **Note:** If you don't see this option, enable Streaming API in your org.

Option	Description
None	Lists don't refresh and push notifications don't appear.
Refresh List	The entire list refreshes when there are any changes to it. Records are added or removed from the list based on the list's criteria, but new records added to queues only refresh the record owner's list.
Refresh List Rows	Rows in the list refresh when there are any changes to fields selected for push notifications.

4. In `Choose How Detail Pages Refresh`, select when push notifications appear.

Option	Description
Do Not Refresh	Detail pages don't refresh and push notifications don't appear.
Automatically Refresh	The detail page automatically refreshes when a record is changed.
Flag	A message appears on the detail page when a record is changed.

5. Click **Select objects and fields for notifications**, and select **Edit**. The push notification settings you choose here apply to all your console apps.
6. Select the objects that you want to trigger push notifications. For example, if you want any changes to cases or case fields to trigger push notifications, move `Cases` from `Available Items` to `Selected Items`.

EDITIONS

Available in Salesforce Classic


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

USER PERMISSIONS

To configure push notifications:

- "Customize Application"

7. Under Fields, click **Edit** and choose the fields you want to trigger push notifications.
8. Click **OK**, then select **Save**.
9. Grant push notification users at least the “Read” permission on the Push Topics standard object. See [User Permissions and Access](#).

 **Important:** When the `Require HttpOnly` attribute is turned on for session security, push notifications don't appear.

 **Note:**

- These objects and their fields are available for push notifications: accounts, contacts, cases, leads, opportunities, campaigns, tasks, and custom objects.
- Push notifications aren't available in the console in Professional Edition.
- If you set up push notifications for the lead object and use Data.com Clean rules, push notifications are triggered in the console. Keep in mind that extra notifications can affect your push notification limit.


Customize Keyboard Shortcuts for a Salesforce Console in Salesforce Classic

Create or customize keyboard shortcuts so that users can quickly perform actions by pressing keys instead of having to use a mouse.


If a console is set up, several default keyboard shortcuts are automatically turned on and available for customization. For example, pressing keys to open and close tabs, move between tabs, and save records. Before you can create custom shortcuts, a developer must define the shortcut's action with the `addEventListener()` method in the Salesforce Console Integration Toolkit. You can only create shortcuts for actions performed inside of a console.

1. From Setup, enter `Apps` in the `Quick Find` box, then select **Apps**.
2. Click **Edit** next to a Salesforce console app.
3. Click **Customize keyboard shortcuts**.
4. Click **Edit** on the top of the page.
5. Click **Edit** to customize an existing shortcut, or click **+Add Keyboard Shortcut** to create a new one. When creating a new shortcut, you must add a `Console Event Name`, which is a unique identifier for the key commands, and then send the event name to the developer who defined the shortcut's action so that he or she can add it to the method that will trigger the shortcut. You can't delete the default shortcuts, but you can edit or deactivate them.
6. Click **Save**.

After you customize or create shortcuts, notify users assigned to a console about the new actions they can perform.

 **Note:** You can have up to 144 shortcuts for each console, which includes the default shortcuts and any that you create. If you create or customize keyboard shortcuts that include standard browser shortcuts, such as CTRL+F, your shortcuts might not work as intended.

Keyboard shortcuts aren't case sensitive, even when you press SHIFT before a letter key. For example, SHIFT+b is the same as SHIFT+B. If CTRL, ALT, or SHIFT is in a keyboard shortcut, the sequence of those keys doesn't matter. For example, CTRL+ALT+A is the same as ALT+CTRL+A.

 **Tip:** You can include as many keys in a shortcut as you want, but we recommend that you limit a shortcut to three keys so that users can remember it. Console users can press SHIFT+K to view an overlay with a list of all console shortcuts.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To customize keyboard shortcuts for a Salesforce console:

- “Customize Application”


Set Up Interaction Logs for a Console in Salesforce Classic

An interaction log lets Salesforce console users write notes on records that appear on primary tabs.

For example, when cases appear on primary tabs, an interaction log can appear so that users can take notes on cases. You can create multiple interaction logs and customize them to display specific task fields for different users to update.


1. From Setup, enter *Interaction Log Layouts* in the **Quick Find** box, then select **Interaction Log Layouts**.
2. Click **New**.
3. Name the interaction log.
4. Select task fields to add to the interaction log and click **Add**.
 - * indicates required fields.
 - You can only add editable task fields to interaction logs.
 - The **Enter your notes here...** field is automatically added to all interaction logs; you can't remove it.
5. If you want this interaction log to be the default for all users, select **Set as default layout**.
You can't delete a default interaction log; you must first mark another interaction log as the default for your organization.
6. Click **Save**.

After you set up or customize interaction logs, you can assign them to different user profiles and turn them on to display.

 **Tip:** You can create custom fields for tasks and add them to interaction logs. For example, you can create a **Caller Disposition** picklist with values of Angry, Neutral, and Satisfied.

Turn On Interaction Logs for a Console in Salesforce Classic

To display interaction logs, you must turn them on for page layouts assigned to different user profiles.

1. Edit any page layout and click **Layout Properties**.
 2. Check **Interaction Log** and click **OK**.
 3. Click **Save**.
-  **Note:** If you turn interaction logs on or off, you must close and reopen records to see the changes.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To set up interaction logs:

- "Customize Application"

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS


To turn on interaction logs:

- "Customize Application"

Assign Interaction Logs in Salesforce Classic

After you set up or customize interaction logs, you can assign them to different user profiles.

1. From Setup, enter *Interaction Log Layouts* in the **Quick Find** box, then select **Interaction Log Layouts**.
2. Click **Log Layout Assignment**.
3. Choose an interaction log to assign to each user profile.
4. Click **Save**.

 **Note:** You can't assign interaction logs to portal user profiles because Salesforce console apps aren't available to portal users.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To assign interaction logs:

- "Customize Application"

Assign the Service Cloud Feature License to Users

The Service Cloud User feature license entitles users to additional Salesforce features, such as the Salesforce console.

To assign the license to a user:

1. From Setup, enter *Users* in the **Quick Find** box, then select **Users**.
2. Click **Edit** next to a user's name.
3. Select **Service Cloud User**.
4. Click **Save**.

EDITIONS

Available in Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited,** and **Developer Editions** with Sales Cloud and Service Cloud

USER PERMISSIONS

To assign the Service Cloud User feature license to users:

- "Manage Internal Users"

Assign the Salesforce Console User Permission Set to Sales Console Users

After you've assigned the "Sales Console User" permission set license to users, assign users to the "Salesforce Console User" permission set.

1. From Setup, enter *Permission Sets* in the *Quick Find* box, then select **Permission Sets**.
2. Select **Salesforce Console User**. This standard permission set is already created for you and comes with the Sales Console permission set license.
3. Click **Manage Assignments** and assign users to the permission set.

You can also choose to clone the "Salesforce Console User" permission set instead and assign Sales Console users to the clone instead.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To create permission sets:

- "Manage Users"

Hide a Salesforce Classic Console from the App Launcher in Lightning Experience

By default, Salesforce Classic consoles appear in the App Launcher in Lightning Experience. This visibility lets your console users access all your consoles apps, ones created in Salesforce Classic and ones created in Lightning Experience, without having to switch UIs. You can change this behavior if you want to limit access to your Salesforce Classic consoles.

1. From Setup, enter *App Menu* in the *Quick Find* box, then select **App Menu**.
2. From the list of available apps, click **Visible in App Launcher** or **Hidden in App Launcher** to show or hide individual apps from the App Launcher in Lightning Experience for all users in the org.

Alternatively, you can also remove access to the Salesforce Classic console for individual users through their profiles.

EDITIONS

Available in Salesforce Classic and Lightning Experience

Available in: **Professional, Enterprise, Performance, Unlimited, and Developer** Editions with Sales Cloud and Service Cloud

USER PERMISSIONS

To view apps:

- "View Setup and Configuration"

To manage apps:

- "Customize Application"
- AND
- "Service Cloud User" OR "Sales Console"


LEARN ABOUT MORE CONSOLE FEATURES

Turn On Multi-Monitor Components for a Salesforce Console in Salesforce Classic

If a console is set up, multi-monitor components are turned on so that you can move portions of a console to different areas of your screens to help you work best.

With multi-monitor components, you can pop out primary tabs or Open CTI SoftPhones from browsers and drag them across multiple monitors. You can also move the History component, Live Agent chats (but not the footer component), and any console components included in a console's footer.

1. From Setup, enter `Apps` in the `Quick Find` box, then select **Apps**.
2. Select a console app.
3. Click **Edit**.
4. If multi-monitor components are turned off, click `Enable Multi-Monitor Components`.
5. Click **Save**.

 **Note:** Popped out items close and lose their positions on screen when you log out of Salesforce, click **Setup**, click a link to leave a console, or refresh or close browsers.

 **Tip:** To pop out components, you might need to update these settings in your browser: (1) allow popup windows; (2) set links to open in new windows, not tabs.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To turn on multi-monitor components:


- "Customize Application"

Save User Sessions for a Salesforce Console in Salesforce Classic

Administrators can save user sessions so that when you close your browser or log out of Salesforce, you can quickly return to what you were working on when you log in again.

If a console is set up, user sessions are automatically saved so that when you log in again, the following appear: any previously open console tabs, custom console components, resized sidebar components, an open or closed search header, and interaction logs.

1. From Setup, enter `Apps` in the `Quick Find` box, then select **Apps**.
2. Select a console app.
3. Click **Edit**.
4. If user sessions aren't saved, click `Save User Sessions`.
5. Click **Save**.

 **Note:** Console uses the browser `sessionStorage` property to retain tabs until a user closes them. If a user logs out of Salesforce and logs in again using the same window, the previously open tabs reopen in the new session, even when `Save User Sessions` is disabled.

EDITIONS

Available in Salesforce Classic

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


USER PERMISSIONS

To save user sessions for a Salesforce console:

- "Customize Application"

Turn On the History Component for a Salesforce Console in Salesforce Classic


If a console is set up, you can add the History component to the footer so that users can quickly access recent primary tabs or bookmarks without having to search Salesforce.

 **Note:** With Summer '15, Most Recent Tabs was renamed History and bookmarks were added to it to give you a more streamlined experience for finding and retrieving tabs. These features aren't available in Internet Explorer® 7 and 8.

1. From Setup, enter *Apps* in the *Quick Find* box, then select **Apps**.
2. Select a Salesforce console app.
3. Click **Edit**.
4. In *Choose Console Components*, add History to *Selected Items*.
5. In *Align Custom Console Component*, choose whether the component will appear in the footer's right or left side.
6. Click **Save**.

Salesforce Call Center

Call Center integrates Salesforce with third-party computer-telephony integration (CTI) systems. Call center users can see Salesforce information for incoming calls, make out-going calls directly from Salesforce, and report on call outcome, duration, and more. A call center is great for boosting productivity for Salesforce users that spend time on the phone.

 **Note:** Keep in mind that the out-of-the-box Service app is a Salesforce Classic app and only supports Open CTI for Salesforce Classic. To make calls in Lightning Experience, use Open CTI for Lightning Experience in a Lightning Experience app, like our out-of-the-box Service Console app.

To set up a call center, work with a developer or partner to create a CTI implementation that uses the Open CTI API and works with your existing telephony system. Most call centers are created by installing an AppExchange package, then all you have to do is decide which users you want to access the call center.

After a call center is set up, call center users can make and receive calls with a softphone. Each softphone looks and behaves differently because each CTI implementation is unique. In the console and in Lightning Experience, softphones appear in a footer. In Salesforce Classic, softphones appear in the left sidebar of every Salesforce page.

The call center is all about customization. You can modify softphone layouts and assign specific layouts to selected user profiles. You can also add phone numbers to call center directories so your users all have access to key phone numbers. As your needs change, your call center can be customized and changed too. As an admin, some customization you can do yourself. However, you might want to work with your developers or partners to make functionality changes.

SEE ALSO:

[Salesforce Help: Set Up a Call Center](#)

[Salesforce Help: Salesforce Open CTI](#)

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To turn on the History component:

- "Customize Application"

EDITIONS

Available in: Salesforce Classic and Lightning Experience

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

Turn On Knowledge for a Salesforce Console in Salesforce Classic

If Salesforce Knowledge is set up, you can turn on the Knowledge sidebar so that users working on cases can automatically see relevant knowledge articles that can help them close cases faster.

! **Important:** You must have Salesforce Knowledge set up in your Salesforce org before you can add it to a console. Knowledge articles appear based on matching words typed into a case's `Subject`.

1. From the object management settings for cases, go to Page Layouts.
2. Click **Edit** next to a page layout.
3. Click **Layout Properties**.
4. Check `Knowledge Sidebar`, and click **OK**.
5. Click **Save**.

Repeat these steps for each case page layout assigned to console users.

SEE ALSO:

[Salesforce Help: Salesforce Knowledge Documentation Overview](#)

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To turn on the Knowledge sidebar:

- "Customize Application"

Add Live Agent to the Salesforce Console in Salesforce Classic

Adding Live Agent to the Salesforce console enables agents and supervisors to chat with customers and access other customer service tools in one place.

Before you add Live Agent to a Salesforce console app, you need to create a Salesforce console app if you don't have one set up.

After you set up Live Agent, add it to a Salesforce console app. After Live Agent is set up in the console, your agents can interact with chat customers. With the Salesforce console, your agents and supervisors can access Live Agent and other Service Cloud products in one place to provide customers fast and efficient customer service.

1. From Setup, enter `Apps` in the `Quick Find` box, then select **Apps**.
2. Click **Edit** next to the name of the Salesforce console app in which you want to set up Live Agent.
3. Select `Include Live Agent in this App`.
4. Choose the records or pages that you want to open as subtabs of chat sessions in the chat workspace.
5. Optionally, if your Salesforce org has Knowledge enabled, select `Include Suggested Articles from Knowledge in Live Agent` to display the Knowledge One widget in the chat workspace.
6. Click **Save**.

EDITIONS

Available in: Salesforce Classic

Live Agent is available in: **Performance** Editions and in **Developer** Edition orgs that were created after June 14, 2012

Live Agent is available in: **Unlimited** Edition with the Service Cloud

Live Agent is available for an additional cost in: **Enterprise** and **Unlimited** Editions

USER PERMISSIONS

To add Live Agent to the Salesforce console:

- "Customize Application"

You can run multiple Salesforce apps at the same time. However, if you log in to another Salesforce app while you're logged in to a Salesforce console app, you can't accept new chat requests.


SEE ALSO:

[Live Agent for Administrators](#)

Macros in the Salesforce Classic Console

Support agents who use Case Feed now can run macros to automatically complete repetitive tasks—such as selecting an email template, sending an email to a customer, and updating the case status—all in a single click. A macro is a set of instructions, which tell the system how to complete a task. When an agent runs a macro, the system performs each instruction. Macros save time and add consistency to support agents' work.

You can create macros to perform multiple actions on the Case Feed. For example, a macro can enter the subject line of an email and update the case status. A single macro can perform multiple actions on different parts of the Case Feed at the same time.

 **Note:** Macros are supported on feed-based standard and custom objects.

SEE ALSO:

[Salesforce Help: Get Started with Macros](#)

EDITIONS

Available in Salesforce Classic

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

SUPER CHARGE YOUR CONSOLE

Customize a Console with Custom Components in Salesforce Classic

Create components to customize, extend, or integrate the Salesforce console with other systems.

Components let you display many types of data in the footer, sidebars, highlights panels, and interaction logs of a console. You can create components to solve a variety of business problems and to display items, such as:

- Third-party apps
- Chat or Softphone widgets
- Content when users view specific pages
- Contextual information alongside records
- Custom highlights panels or interaction logs
- Content when users click a customizable button in the footer

Components let you get creative with a console and build what your users need to do their jobs. To create components, you can use Visualforce, canvas apps, lookup fields, related lists, or report charts. You can also use Force.com Canvas or the Salesforce Console Integration Toolkit to build components. Both technologies are JavaScript APIs that let developers extend or integrate a console. The toolkit provides you with programmatic access to a console to do things like open and close tabs to streamline business processes, whereas Canvas helps you integrate a console with external applications that require authentication methods.

For more information about canvas apps or the toolkit, see [Force.com Canvas Developer's Guide](#) and the [Salesforce Console Integration Toolkit Developer's Guide](#).

EDITIONS

Available in Salesforce Classic

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

Process for Creating Console Components in Salesforce Classic

Developers and administrators typically follow these steps when creating components for a Salesforce console.

Even though you can create a variety of console components to solve many types of business problems, the process for creating components generally follows these steps.

1. Developers create a Visualforce page or a canvas app.
2. Administrators add the Visualforce page or canvas app to either:
 - Page layouts to display content on specific pages or
 - Console apps to display content across all pages and tabs
3. Administrators who add a component to page layouts:
 - Choose the sidebar in which the console displays, such as right, left, top, or bottom
 - Select whether to add multiple components to a sidebar and arrange them as either a stack, tabs, or accordion
4. Administrators who add a component to apps:
 - Specify the look and feel of the button used to access the component from the console's footer
 - Assign the component to each console where they want users to access it

EDITIONS

Available in Salesforce Classic

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

- Users view the component when they access page layouts or click a button in the footer of a console.



Tip: To add a lookup field or related list as a component, you can do so without creating a Visualforce page or canvas app. Just follow the steps in “Add Console Components to Page Layouts in Salesforce Classic,” and in the `Type` field, select Lookup or Related List to choose a specific field or list.

Console Component Implementation Tips in Salesforce Classic

Before you create a component for a Salesforce console, review these tips.

Behavior and Access Tips

- You can't add canvas app components to AppExchange packages.
- When you update records on primary tabs or subtabs, components don't refresh automatically.
- If you change the `Height` or `Width` of a console component, refresh your browser to see your updates.
- Report chart components aren't available to developers using the Salesforce Metadata API or for migration between sandbox and production instances.

EDITIONS


Available in Salesforce Classic

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

Page Layout Tips

- You can add components to page layouts if you're assigned to a Sales Cloud User Permission or Service Cloud User feature license.
- After you turn off the highlights panel or interaction log on a page layout, you can add a component to the Top Sidebar or Bottom Sidebar of primary tabs.
- If you want a component to display across all the subtabs on a primary tab, add the component to the Primary Tab Components section on page layouts. If you add a component to the Subtab Components section on page layouts, the component displays on individual subtabs.
- If you edit a page layout in the console, such as adding a primary tab component like an interaction log, the updated layout isn't displayed until the page is closed and then reopened in a new primary tab.
- If [Salesforce Knowledge is enabled](#) and the [Knowledge sidebar is turned on](#), don't add a component to the Right Sidebar of case page layouts, otherwise the Knowledge sidebar displays over components.

Visualforce Tips

- For each page layout, you can use a Visualforce page as a component once.
- If Visualforce pages are assigned to page layouts as components, you can't delete them.
- Visualforce pages don't refresh when you click  and select **Refresh all primary tabs** or **Refresh all subtabs**.
- If you enable clickjack protection for Visualforce pages in your organization's security settings, Visualforce pages won't display correctly.
- Unlike other Visualforce pages, you don't have to set the standard controller on components to the object whose page layout you're customizing.
- If you add `showHeader=true` to a Visualforce page, you can add a 15-pixel gap to the right and left sides of a component to visually indicate its location in a sidebar. For example, `<apex:page standardController="Contact" showHeader="false" title="List of Cases">`.
- When Visualforce pages are used as components, two URL parameter values are passed automatically:

Parameter Name	Parameter Value	Description
<code>id</code>	A case-sensitive 15-character alphanumeric string that uniquely identifies a record.	The ID of the detail page in which the component displays. This parameter is only passed if the ID is available.
<code>inContextPane</code>	<code>true</code>	Indicates the Visualforce page displays as a component in a Salesforce console.

Add Console Components to Page Layouts in Salesforce Classic

After you create a console component, you can add it to page layouts so that users can access it from a sidebar in a Salesforce console.

- From the object management settings for the object whose page layout you want to edit, go to Page Layouts.
- Click **Edit** next to a page layout.
- Click **Custom Console Components**.
- Next to the sidebar in which you want to add your component, choose how to display and organize your components by `Style`. See [Sidebar Styles for Console Components in Salesforce Classic](#).
- Specify the size of the sidebar in the console by selecting height or width, and typing its percentage or number of pixels. When the height or width of the sidebar is below the following number of pixels, these components types automatically hide because they become too difficult to read:
 - Knowledge One - 250
 - Lookups - 150
 - Related lists - 200
 - Report charts - 150
- Under `Type`, choose whether your component is a Visualforce Page, Canvas App, Lookup, Related List, or Report Chart. Before you can choose a Report Chart, add it to the page layout and make sure it's in a folder that's shared with users who need access.
 - If the Knowledge tab is set up in your Salesforce org, add the Knowledge One component to case pages so agents can easily find and work with articles.
 - If your org uses entitlement management, add the Milestone Tracker to case pages so agents can see the latest information on milestones related to cases they're working with.
 - If you have topics enabled for an object, add the Topics component so users can view, add, and delete topics on the records they're working with.
 - Add the Files component to case pages so support agents can use drag-and-drop to quickly attach files to cases, see files that are already attached, and include attached files to case emails.
 - If your org uses Knowledgeable People and topics on objects, add the Case Experts component to case page layouts so agents can easily collaborate with colleagues who have expertise in specific areas.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To add custom console components:

- "Customize Application"

AND


"Service Cloud User" OR
"Sales Console"

To create Visualforce pages:

- "Customize Application"

To manage canvas apps:


- "Customize Application"

 **Note:** Case Experts is available through a pilot program. For information on enabling Case Experts for your org, contact Salesforce.

Since page layouts are limited to two report charts, you can add up to two report charts to a sidebar. You can add up to five components to a sidebar and a maximum of eight components to a page layout. Any canvas app components you add can't be added to AppExchange packages.

7. Under **Component**, set the details:

- For Visualforce pages or canvas apps, click  under **Component** to find and select your component.

 **Tip:** You can include Lightning components in Visualforce pages. See [Lightning FAQ](#).

- If your component is a lookup field, related list, or report chart, choose the field, list, or chart's name. Optionally, you can add more fields to look up components by defining objects' mini page layouts. You can also hide report charts and related lists from pages where they appear as components to prevent duplicate information from showing. If you want to reorder related lists, do so from page layouts. Related list components don't display custom buttons.

8. Under **Label**, type the name for your component to display to console users from the sidebar. If you don't type a label for a report chart, the report chart's name is used.

9. Specify the size of the component in the sidebar by selecting height or width, and typing its percentage or number of pixels. If you don't specify a height or width, the component is automatically sized to appear correctly in a console.

Autosize Components is available for components with the **Stack** style. It's selected by default so that when users resize browsers, **Stack** components fit nicely in sidebars. Salesforce chooses a minimum height and width for **Stack** components, and you can't change those values. The **fit-automatically** option is unavailable for Visualforce pages, and canvas apps.

10. Click **Save**.

Add Quick Actions to the Console Sidebar in Salesforce Classic

Add custom quick actions to the console sidebar to let support agents easily create, update, and search for records associated with cases. Quick actions are added as console components.

Create the quick actions before adding them to the Case page layouts in the console.

1. From Setup, enter *Page Layouts* in the **Quick Find** box, then select **Page Layouts** under **Cases**.
2. Select a page layout to work with.
3. Click **Custom Console Components**.
4. Scroll to the sidebar section where you want to add the quick action component. For example, to add it to the left sidebar, go to the **Left Sidebar** section.
5. Configure the quick action component:
 - **Field:** Select the record field that the quick action affects. For example, if the quick action creates a contact record, select the **Contact** field.
 - **Enable Linking:** If selected, support agents can associate this record with a case. For example, a support agent can create a contact and link it to a case.
 - **Create Action:** Select the "Create" quick action. This quick action lets support agents create a record.
 - **Update Action:** Select the "Update" quick action. This quick action lets support agents update a record.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To create, edit, and delete actions and page layouts, and to add custom console components:

- "Customize Application"

- 6. Add more quick action components as needed.
- 7. Click **Save**.

Sidebar Styles for Console Components in Salesforce Classic




When administrators add components to page layouts, they choose the sidebar style in which to display the components in a Salesforce console.

The following sidebar styles are available for console components on page layouts.

EDITIONS


Available in Salesforce Classic


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

Multi-Component Type	Description
Stack	A content area with multiple frames. 
Tabs	A single content area with a list of multiple panels. 
Accordion	A collapsible content area.  This style is only available for vertical sidebars.

Add Console Components to Apps in Salesforce Classic

After you create a console component, you can add it as a button in the footer of Salesforce console so users have access to it.

1. From Setup, enter *Custom Console Components* in the **Quick Find** box, then select **Custom Console Components**.
2. Click **New**.
3. Type a name for your component.
4. In **Button Name**, type the label to display on the button users click to launch your component. For example, *Live Agent*.
5. In **Button CSS**, enter the in-line style used to define how the button looks to users who click it to launch your component. The styles supported include font, font color, and background color.
6. Type the width of the button to display in the console.
7. In **Component**, select **Visualforce Page** or **Canvas App**, and either type the name of your component, or click  to find and select it.

 **Tip:** You can include Lightning components in Visualforce pages. See [Lightning FAQ](#).
8. Type the height and width of the window used to display your component in the console.
9. To prevent users from changing the dimensions of the window used to display your component, click **Fixed Width** or **Fixed Height**.
10. Click **Save**.

Before users can access your component, you must assign it to a console. See [Assigning a Custom Console Component to an App](#).

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To add custom console components:

- "Customize Application"

AND

"Service Cloud User" OR "Sales Console"

To create Visualforce pages:

- "Customize Application"

To manage canvas apps:

- "Customize Application"

Assign a Console Component to an App in Salesforce Classic

After you add a console component to Salesforce console apps, you must assign it to each console for user access.

1. From Setup, enter *Apps* in the **Quick Find** box, then select **Apps**.
2. Select a console app.
3. Click **Edit**.
4. In **Choose Custom Console Components**, add any available components to your app.
1. In **Align Console Components**, choose whether to align components to the left or right of the console's footer.
5. Click **Save**.

More Resources

To take your console implementation to the next level, check out these design guidelines and best practices.

EDITIONS

Available in Salesforce Classic

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

USER PERMISSIONS

To assign custom console components to an app:

- "Customize Application"

- [*Service Console Design Guide*](#)
- [*Salesforce Console Best Practices: Creating a Desktop Your Agents Love*](#)

INDEX

A

Audience [1, 28](#)

C

Call Center

about [20](#)

Configurable features

list of [6](#)

Configure Salesforce console

highlights panel [11](#)

interaction logs [16–17](#)

keyboard shortcuts [15](#)

Knowledge sidebar [21](#)

list of configurable features [6](#)

multi-monitor components [19](#)

pinned lists [12](#)

push notifications [14](#)

save user sessions [19](#)

whitelist domains [13](#)

console

quick actions [26](#)

CTI

See Call Center [20](#)

Custom console components

add sidebars [25, 27](#)

add to apps [28](#)

add to page layouts [25, 27](#)

assign to consoles [28](#)

Canvas app [25](#)

creation overview [23](#)

implementation tips [24](#)

Knowledge One [25](#)

lookup [25](#)

multiple components [25](#)

overview [23](#)

related list [25](#)

report charts [25](#)

Visualforce page [25](#)

H

Highlights panel

customize [11](#)

History

Salesforce console [20](#)

Hot keys

configure [15](#)

I

Implementation tips [3](#)

Interaction log

assign [17](#)

set up [16](#)

turn on [16](#)

K

Keyboard shortcuts

configure [15](#)

Knowledge One

console component [25](#)

Knowledge sidebar

turn on for Salesforce console [21](#)

L

Limitations [4](#)

Live Agent

add to the Salesforce console [21](#)

pop out chats [19](#)

Lookup components [23](#)

M

macros [22](#)

Multiple monitors [19](#)

O

Overview

Call Center [20](#)

P

Pinned lists

configure [12](#)

Planning

questions [2](#)

Pop out components

turn on [19](#)

Push notifications

configure [14](#)

Q

quick actions [26](#)

S

- Sales Console user permission
 - enable [18](#)
- Salesforce console
 - about [1](#)
 - access [18](#)
 - App Launcher [18](#)
 - component [20](#)
 - copy a Salesforce console app [9](#)
 - copy a Service Cloud Console app [9](#)
 - create a Salesforce console app [9](#)
 - create a Service Cloud Console app [9](#)
 - customize with custom console components [23](#)
 - for sales [1](#)
 - for service [1](#)
 - history [20](#)
 - limitations [4](#)
 - most recent tabs [20](#)
 - save user sessions [19](#)
 - set up [9](#)
 - set up a Salesforce console app [9](#)
- Salesforce Console
 - list view hovers [12](#)
 - responsive lists [12](#)
- Salesforce Console Integration Toolkit
 - creating custom console components with [23](#)
- Service Cloud User license
 - assign [17](#)
- Set up Salesforce console
 - assign Service Cloud User feature license [17](#)
 - colors [10](#)
 - enable Sales Console user permission [18](#)
 - footer [10](#)

Set up Salesforce console (*continued*)

- header [10](#)
- highlights panel [11](#)
- implementation tips [3](#)
- interaction logs [16](#)
- overview [2](#)
- sales console setup [7](#)
- service console setup [8](#)

Sidebars

- creating [25](#)
- customizing [25](#)
- Knowledge sidebar [21](#)
- position [28](#)

SoftPhone

- pop out [19](#)

T

Tabs

- history [20](#)
- most recent [20](#)
- pop out [19](#)

U

User sessions

- save for a Salesforce console [19](#)

V

Visualforce components

- add to a Salesforce Console [23](#), [28](#)

W

Whitelist

- domains [13](#)