

PUBLIC
Document Version: 8.10 rev 12 – 2026-05-22

SAP GUI for the Java Environment

© 2026 SAP SE or an SAP affiliate company. All rights reserved.

Content

1 Introduction. . . . 4

2 Installing or removing an SAP GUI for Java version. . . . 6

3 Getting Started. . . . 8

3.1 Configure Connections for SAP Logon. . . . 9

3.2 Add a new Connection in SAP Logon. . . . 10

3.3 Using Direct Connections. . . . 14

4 Working with SAP GUI for Java. . . . 16

4.1 SAP Logon Window. . . . 16

Additional Functions in SAP Logon. . . . 18

4.2 Preferences Dialog. . . . 20

Configuration. . . . 20

Design. . . . 21

Desktop. . . . 24

Other. . . . 25

Web AS ABAP. . . . 27

4.3 Elements of an SAP GUI for Java Session Window. . . . 31

5 Reference. . . . 35

5.1 Installation Reference. . . . 35

5.2 Configuration Files. . . . 40

Centrally Managed Configurations. . . . 41

Local Configuration Files. . . . 42

Installation of Custom Data. . . . 42

Overview on Local/Central Files. . . . 44

5.3 Security Policy. . . . 44

Concepts and Implementation. . . . 45

Policy Syntax. . . . 47

Default Trust Levels. . . . 50

Bibliography. . . . 51

5.4 SNC Encryption . . . 51

5.5 Certificates Management. . . . 52

5.6 Accessibility. . . . 52

5.7 Connection Strings. . . . 53

5.8 Connection Parameters. . . . 56

5.9 Scripting. . . . 58

5.10 Trace Information. 58

6 Appendix. 63

6.1 Miscellaneous Product Information. 63

6.2 Further Information Sources. 64

1 Introduction

Platform Independence

The SAP GUI for the Java Environment (SAP GUI for Java) is a unified SAP front end running on Linux, macOS and Windows for connecting to Web Application Server ABAP (Web AS ABAP).

The technology behind SAP GUI for Java is a combination of Java and C/C++ running with a platform independent code base.

All the rendering is done in Java using Swing components to provide highly portable and consistent functionality on all supported platforms. The corresponding components of the standard ActiveX controls used in SAP GUI for Windows are reimplemented as JavaBeans. Already existing C/C++ libraries on the respective platforms are used for functions such as network communication, protocol handling, or RFC.

Various SAP branded themes like Morning Horizon, Evening Horizon, Quartz Light, Quartz Dark, Quartz High Contrast Black or Quartz High Contrast White are available as Swing Look and Feels to provide a consistent user experience for all UI elements and platforms.

Deployment

SAP GUI for Java can be simply deployed by running the native installation programs, see chapter Installing or removing an SAP GUI for Java version [page 6]. It can also deployed by running the Java installation package. It is possible to run the installation in unattended mode using a response file containing all necessary information and options. For detailed information, see topic Installation Reference [page 35].

Security

The Java Security Model is employed to provide protection mechanisms for using features of the desktop, like file system access, printing or executing processes.

Centralized Configuration

SAP GUI for Java needs configuration information about your SAP environment, such as the names and addresses of your SAP servers. Based on this information, a connection directory is offered which contains all available connections that can be selected in the SAP Logon. This directory can be centrally stored on a web server and only a URL needs to be configured in SAP GUI for Java. Preset configuration and options can be

distributed as templates during the initial installation process, so that a manual configuration after the first installation of SAP GUI for Java is not required.

Scripting

SAP GUI for the Java Environment can be automated by executing scripts that emulate user interaction. For more information, see Scripting [page 58].

2 Installing or removing an SAP GUI for Java version

> **Note**
>
> For detailed information about the installer commands and options see chapter Installation Reference [page 35]. There are specific options to influence the installation procedure for a manual or remote installation as administrator for many clients.

You can install or remove an SAP GUI for Java version either using the platform-dependent, native installation program (see below), or using the installation package (`PlatinGUI-<platform>-<version>.jar`). Each installation program or installation package relates to a specific SAP GUI for Java version.

> **Note**
>
> If you use the installation package, you will need an installed Java Virtual Machine. For information about using the installation package, see chapter Installation Reference [page 35].

Platform-dependent Installation Programs

The platform-dependent installation programs use the inbuilt SapMachine during the installation, thus no installed Java Virtual Machine is required. Depending on your platform, the following files are applicable:

- Linux platform: PlatinGUI-Linux-Installation-<version>
- macOS platform: PlatinGUI-MacOSX-Installation-<version>.dmg
  The dmg for macOS does not contain an installer, but a ready-to-copy application. Alternatively, the jar installer can be used in case the following installation features are required:
  - the need of Installation options
  - the deployment of templates to preconfigure SAP GUI for Java, see Use of templates to preconfigure a SAP GUI for Java Installation [page 43]
  - the installation of External Installation Packages [page 38]
- Windows platform: PlatinGUI-Windows-Installation-<version>.exe

Prerequisites

- To download a SAP GUI for Java installation program from the SAP Service Marketplace you need a user with download authorizations. If you do not already own a corresponding user, follow this link to request a user ID. For more information regarding software download authorizations, see SAP Note 1037574.
- Check the Miscellaneous Product Information [page 63].

Installation Procedure using the Installation Program

To install or remove an SAP GUI for Java version using the installation program, follow the instructions below for a simple and dialog based procedure.

1. Go to the SAP GUI Family homepage. From the table in the **Downloads** section choose the *Patches* link for the current SAP GUI for Java release and choose the appropriate installation program for the operating system of your client PC.

2. Run the downloaded installation program.
   The installer window opens.
3. The installation is a guided procedure, you can follow the instructions you get from the tool:
   1. **Introduction**
      Read the information and choose *Next*.
   2. **Readme**
      Read the information and choose *Next*.
   3. **Define Options**
      Change the default options only if necessary and choose *Install* to start the installation process.

      > **Note**
      >
      > If the installation program detects that this version of the SAP GUI for Java is already installed, you will have the options to remove or to (re-)install the installation.

   4. **Install**
      SAP GUI for Java will be installed. (In case you have chosen *Remove*, this version of SAP GUI for Java will be removed.)
   5. **Summary**
      Check the installation details if necessary and close the dialog.

3 Getting Started

You can connect to an SAP System by starting the SAP Logon window and choosing the SAP System from the list, or by starting a direct connection to the server.

Starting SAP Logon on Windows

You have the following options to start SAP Logon on Windows:

- Choose *Start* > *SAP Clients* > *SAP GUI for Java* from the Windows Start menu.
- If the desktop icon is available, you can also start SAP GUI for Java using its desktop icon. The option *Install Desktop Icon* is set per default at the installation, see also chapter Installing or removing an SAP GUI for Java version [page 6].
- You can run the **guilogon.bat** file in the `C:\...\SAP Clients\SAP GUI for Java <version>\bin` folder. (Note, that you can change the default path during the installation.)

The SAP Logon window appears. It shows a list of connections to SAP Systems.

> ⓘ Note
>
> If the list of connections is empty at start-up, make sure that your local configuration is correct (see chapter Configure Connections for SAP Logon [page 9]).

To connect to a SAP System, choose the related connection from the list.

Starting SAP Logon on Linux

To start the SAP Logon on Linux, invoke the script **`guilogon`** from the GUI installation directory without arguments.

The SAP Logon window appears. It shows a list of connections to SAP Systems.

> ⓘ Note
>
> If the list of connections is empty at start-up, make sure your local configuration is correct (see chapter Configure Connections for SAP Logon [page 9]).

To connect to a SAP System, choose the related connection from the list.

Starting SAP Logon on macOS

In the Finder of macOS, choose `/Applications/SAP Clients/SAPGUI <version>` folder and run the *SAPGUI <version>* program file.

The SAP Logon window appears. It shows a list of connections to SAP Systems.

> **Note**
>
> If the list of connections is empty at start-up, make sure your local configuration is correct (see chapter Configure Connections for SAP Logon [page 9]).

To connect to a SAP System, choose the related connection from the list.

3.1 Configure Connections for SAP Logon

SAP GUI for Java needs configuration information about your SAP environment, such as the server names and addresses of your SAP Systems. Based on this information, a connection directory is offered which contains all available connections displayed in the SAP Logon list. The necessary information is usually prepared by your system administrator, who sets up the configuration files. A configuration file can either be distributed to each front-end computer, or it can be kept centrally on an intranet web server, where all front-end computers can access it. For more information, see chapter Configuration Files [page 40].

Logon Configuration in Preferences

In the SAP Logon window, choose *File* > *Preferences* to open the *Preferences* window. In the *Preferences* window choose *Configuration* > *Logon*. The following options are available:

- *General* > *Configuration File*
  You can enter the full path or the URL to the local or the central *Configuration File*. If the configuration file includes the SAP UI Landscape file, its path or URL is displayed in the *SAP UI Landscape* field below and no further entries might be needed.
- *Web AS* > *SAP UI Landscape*
  The *SAP UI Landscape* field displays the path or the URL to a local or the central SAP UI Landscape file. The SAP UI Landscape file provides all relevant information to create connections that are displayed in the SAP Logon list. It can contain some predefined connections or further added connections (see Add a new Connection in SAP Logon [page 10]).

  > **Note**
  >
  > The *SAP UI Landscape* file can already be included by the configuration file. For more information, refer to SAP UI Landscape Configuration Guide.

  In the text output of the *SAP UI Landscape*, you check which configuration of the SAP UI Landscape has been loaded.

- You can switch on and off expert mode. If it is switched on, you can open an SAP UI Landscape file in SAP Logon for preparation, that is, a central SAP UI Landscape file. Go to *File* > *Open Landscape*, *Save Landscape*, and *Save Landscape As...*.
- You can switch on and off a cache which is used to check whether a configuration file has been changed. Only changes of a configuration is reloaded.
- If some connections use local definitions of message servers or routers, and these definitions also exist in the central configuration file, you can remove the local definitions with the *Remove local overwrites* button at the bottom of the Logon Configuration options.

3.2 Add a new Connection in SAP Logon

First Steps: Add a new Connection to an SAP System

SAP GUI for Java displays the connection data to SAP Systems (connections) in a list within the SAP Logon window. This list is empty when you open the SAP Logon window for the first time, unless your system administrator has preconfigured some connections (see chapter Centrally Managed Configurations [page 41]).

To add a connection to the list within the SAP Logon window, choose the *New* button or alternatively *File* > *New* > *New Connection*. In the *Connection Properties* dialog proceed as follows:

1. On the *System* tab, choose an SAP System ID from the drop-down list of the *System* input field. After choosing a System ID, the description and the group/server data will usually be inserted.
2. If required choose another logon group or application server for this System ID from the *Group/Server* drop-down list.

> **Note**
>
> Whenever you are not sure which group/server to use, contact your system administrator. This information is necessary to optimize load balancing.

3. If desired adapt the descriptive system name in the *Description* input field. The descriptive name you enter here for the connection will appear in the system list within the SAP Logon window.
4. *Save* the new connection.

To edit a connection from your SAP Logon list, select it and choose the *Edit* button. To delete a connection, select it and choose the *Delete* button or alternatively *Edit* > *Delete*.

> **Note**
>
> You cannot edit or delete connections that have been preconfigured by your system administrator.

The Connection Properties in Detail

The *Connection Properties* dialog contains besides the above mentioned *Description* the following tabs:

- The System Tab [page 11]
- The Logon Tab [page 11]
- The Security Tab [page 12]
- The Language Tab [page 12]
- The Advanced Settings Tab [page 13]

The System Tab

The *System* tab defines the network connection to the SAP System. It offers the following configuration options:

- **System:**
  First choose an SAP System ID from the drop-down list. The system's message server will then be asked automatically for a list of all groups and servers that are available for connecting to the chosen system. The result will be displayed in the *Group/Server* list. Note that this might take some time depending on the network connection to the message server.
  If the *Group/Server* list is empty, make sure that your message server list is configured correctly.

  > ⓘ Note
  >
  > For specifying a direct connection to an application server, open the *Advanced* tab and activate the *Expert Mode*. For the syntax of the **conn** parameter, refer to Connection Parameters [page 56].

- **Group/Server:**
  Choose a logon group or an application server from the list. The choice of groups and servers is only possible if a valid SAP System ID has been chosen and if this system's message server has been contacted successfully.
- **Router:**
  If you want to reach a remote SAP System using SAP router, select one from this list. Otherwise select *(none)*. Note that the router settings will also affect communication with the message server as described above. If the list is empty and it is required to use a router, make sure your router list is configured correctly.
- **Speed:**
  Mark the *Low Speed Connection* check box if you are using a slow WAN connection for this SAP System. The server will then try to optimize network communication by not transferring data before it is actually requested and by omitting expendable data (for example, decorative images).

The Logon Tab

The *Logon* tab defines logon data that is automatically transferred to the SAP System's logon screen when the connection is established. These fields correspond directly to the fields on the logon screen. Empty fields are ignored.

- **Client:**
  SAP client for this connection.
- **User:**
  SAP user name for this connection.
- **Language:**
  Two-digit letter language code for this connection (for example, EN for English).
- **Transaction:**
  Enter a transaction code to start directly with this transaction instead of the default start screen.

The Security Tab

The *Security* tab defines the Trust Level and the settings for Secure Network Communication (SNC).

> **Note**
>
> You can only use Secure Network Communication if an SNC compliant security product is installed both on your front-end computer and on the SAP server. For more information, see https://help.sap.com/docs/SAP_SINGLE_SIGN-ON.

- **Trust Level**
  Choose the trust level to be used for this Web AS ABAP system. Your choice will affect all connections with this SAP System ID. For information on trust levels in general and the build-in trust levels, see Security Policy [page 44].
- **Enable Secure Network Communication**
  Mark this check box if you want to use SNC for this SAP System. If the check box is not marked, SNC will not be supported by the SAP server.
- **SNC Name**
  The SAP System's SNC name is defined by the message server and is displayed here for your information.
- **SNC quality of protection**
  Select one of the following check boxes:
  - *Authentication*
    Identity of the communication partners is verified.
  - *Integrity*
    Integrity of the transmitted data content is verified.
  - *Encryption*
    Transmitted data content is encrypted.
  - *Max. Available*
    Automatically use best quality available (default).
  - *Use Manual Login (no SSO)*
    SNC is only used to encrypt the transmitted data, but not for Single-Sign-On.

The Language Tab

The *Language* tab defines the language and the code page settings for the configured connections.

The application server needs to know the character encoding (ISO 88951-x, Shift-JIS or whatever else) it has to use to communicate with SAP System. This is done by defining a code page number when opening the connection to the server. This code page depends on the language used when logging on to the server.

The Java Virtual Machine (JVM) itself needs to know about its language environment to set up the fonts needed to display the characters for different languages. This information is required on start-up of the JVM and therefore has to be set before starting SAP GUI for Java.

In summary you should have SAP GUI for Java running in the system locale for your primary language (this defines which set of characters your JVM will display), then you can define the language you want to use when communicating to the application server of an SAP System.

> ⓘ Note
>
> The defaults are always taken from your system locale.

- **Encoding Definition:**
  Application servers support three possible strategies for finding the correct SAP code page.
  1. *Automatic*:
     The application server deduces the appropriate code page from the language and the client operating system given when logging on to the Web AS ABAP.
  2. *Custom*:
     If you want to override this automatic code page selection, you can choose the code page using the drop-down list of the *Language* combo box. To override the proposed code page you can specify the code page number yourself. This is not recommended but may be necessary in rare circumstances (for example, using custom code pages).
  3. *Unicode*:
     If you connect to a Unicode system you might consider using Unicode for the communication. You still need to specify a language and code page since the application server needs a fallback code page for communicating to non-Unicode systems and legacy programs. This code page has to be correct because the application server has no possibility to correct errors in this scenario.
- **Language:**
  You can choose the language you want to use here. This combo box is just for your convenience. You can select the information needed by language and do not have to look up the correct code page for your language. Choosing a language will automatically fill in the values for the code page and the encoding.
- **Code Page:**
  You can manually set the SAP code page number. This setting will be used for communicating with the Web AS ABAP server.
- **Java Encoding:**
  The Java character encoding corresponding to the SAP code page. You can only enter something in this field if the SAP code page has no matching encoding built in. This is an extremely rare case and should not happen.

The Advanced Settings Tab

The *Advanced* tab displays the technical connection information [page 56] as it results from the information in all other fields. It is provided for informational purposes and should only be changed by expert users. Changes made to this field will override all information specified in other fields.

Expert Mode

Activate the *Expert Mode* check box to edit the connection data for this connection manually.

> **Note**
>
> If you deactivate the *Expert Mode* check box after changing the connection data manually, your manual changes will be lost and the edit field will again reflect the information configured in all other fields.

3.3 Using Direct Connections

To start a direct connection to an SAP System, you will need the connection data [page 56] of the SAP System. Note that the connection data is also displayed in the Advanced [page 13] tab of the *Connection Properties* dialog when editing system connections.

The connection data can be stored in a file.

> **Note**
>
> You can save a single connection as a **.sapc** or as a **.sap** file. If you want to exchange connection data between SAP GUI for Java clients, you can use the **.sapc** file type. Only if you exchange connection data between an SAP GUI for Java client and an SAP GUI for Windows client or vice versa, you need to use the **.sap** file type.

Saving Connection Data as Document

You can save the connection data from the SAP Logon window or from the session window [page 31] of an open connection.

1. Start SAP Logon or connect to a server.
2. Choose *File* > *Save Connection Data as Document*.
3. Choose a location and file name in the Save dialog.
4. Save the file as **`SAP Connection Documents (.sapc)`** file (or as **`SAP GUI for Windows Shortcut (.sap)`** file, see Note above).

> **Note**
>
> You can also drag and drop a connection item from the SAP Logon list to your platform-related file explorer.

Editing a Connection String

You can edit local SAP Connection Documents (**.sapc** or **.sap** file) with a suitable editor and change the connection data accordingly.

> ### Example
>
> (.sapc file for the server appserver.acme.com, port 3200):
>
> ```
> conn=/H/appserver.acme.com/S/3200
> ```

For more information see Connection Strings [page 53].

Opening a Connection Data Document

You can open a direct connection using a connection data document (**.sapc** file or **.sap** file) as follows:

- In SAP Logon window or in the session window of an open connection, choose *File* > *Open Connection Data Document* and open the **SAP Connection Documents (.sapc, .sap)** file.
- Drag and drop the connection data document onto the connection list of the SAP Logon window, select the new connection item and choose *Connect*.
- Drag and drop the connection data document onto the *Connect* button of the SAP Logon window.
- Run the connection data document directly. In this case the **.sapc** file type needs to be associated with the SAP GUI for Java program.

4 Working with SAP GUI for Java

4.1 SAP Logon Window

Overview

The SAP Logon window is the starting point of SAP GUI for Java where connections to SAP Systems are maintained and launched. The following functions are available on the menu bar of SAP Logon:

Entry	Subentry
File	• New Connection [page 10] • *Open Recent* Displays a list of recently used connections. • *Open Connection Data Document* The connection data can be stored in a file, which can be created manually or from the menu bar via *File* > *Save Connection Data as Document* when connected to a system. For more information, see Using Direct Connections [page 14]. • *Open Workspace List* In the *Workspace* List you can add or remove your Workspaces. See also chapter Additional Functions. • *Open Message Server List* In the *Message Server* List you can add or remove Message Servers. • *Open Router List* In the *Router* List you can add or remove Routers. • *Close SAP Logon* Closes SAP Logon while existing SAP GUI for Java sessions stay open. • Save Connection Data as Document The connection data can be stored in a file. For more information, see Using Direct Connections [page 14]. • Import Connections You can import connections from a SAP UI Landscape file (.xml). • Export Connections You can export connections to a SAP UI Landscape file (.xml). • Preferences [page 20] • Trace [page 58] • *Exit Application* SAP Logon and all SAP GUI for Java sessions will be closed. If at least one session window is open, a confirmation dialog will be displayed.

Entry	Subentry
Edit	Contains some basic editing functions like *Undo*, *Redo*, *Cut*, *Copy*, *Paste*, etc.
View	There are three different view types available: • *Hierarchy* [page 18] • *Explorer* [page 18] • *List* [page 18] Additionally, the entries in *Explorer* and *List* views and can be displayed with tiles or with columns when the corresponding option is chosen: • *List with Tiles* • *List with Columns* Further customization is possible in all three views: • *Show Memo* [page 18] • *Show Hidden Entries* [page 19] • *Show All Workspaces* [page 19]
Only visible if SAP Screen Personas is enabled and available: *Personas* **Note** In SAP GUI for Java on macOS the Personas menu is always visible.	Contains some functions of SAP Screen Personas. These functions are only enabled if SAP Screen Personas is active.
Scripts	A macro-like interface for the creation and execution of scripts.
Window	Contains some functions for window management. E. g. *Minimize*, *Zoom*, *Switch Window* and a list of all active sessions.
Help	Provides various forms of online help: https://help.sap.com/viewer/product/sap_gui_for_java/770.00SAP GUI for Java Product Page SAP.COM SAP Community Network (SCN) SAP GUI Community SAP Support Portal

The *File*, *Edit*, *Personas* (if available), *Scripts*, *Window* and *Help* menus from the menu bar of SAP Logon mainly correspond to their counterpart inside a session window, except that the *File* menu is extended by some system specific functions.

Maintaining Connections

For modifing the list of connections the buttons *New*, *Edit*, *Duplicate* and *Delete* or a menu item of the context menu [page 20] on an item in the list of connections can be used. More information can be found in chapter Add a new Connection in SAP Logon [page 10].

Displaying Connections

View Types

There are three different view types available.

- *Hierarchy*
  Connections are displayed in a folder hierarchy. Above the hierarchy, there sits a list of favorites, which can be organzised in a hierarchical structure as well. For modifiying the hierarchy of connections or favorites, drag and drop can be used as well as entries like *New Folder* from the context menu [page 20].
- *Explorer*
  This view consists of two parts. On the left hand side there is a folder structure and on the right hand side all connections that belong to the currently selected folder are displayed.
- *List*
  A sequence of all connections in alphabetical order is displayed. You can choose between a column view and tile view.

View Options

When displaying the view type *Explorer* or *List*, there is the option to display the connections as a *List with Tiles* or as a *List with Columns*. Latter can display following columns:

- Name of the connection
- Description of the SAP system
- Connection Strings [page 53]
- Info: an icon to identify if the connection has been created locally or comes pre-configured in a central configuration file [page 41].

All columns except the name can be hidden or shown using the context menu [page 20].

4.1.1 Additional Functions in SAP Logon

Memo

An arbitrary text can be assigned to each connection, for example for describing the purpose of that system or steps of a task executed in that system. Existing memos are indicated by a text icon next to the entry. To display a memo, choose *View* > *Show Memo* in the SAP Logon menu.

Workspaces

Workspaces are an hierarchical level to manage a set of connections above a structure of folders and subfolders. To edit or create a Workspace choose *File* > *Open Workspace List* in the SAP Logon menu. The Workspaces dialog offers the following customization options:

- *New*
  Creates a new workspace.
- *Delete*
  Deletes an existing workspace.

  > **Note**
  >
  > Only user-created workspaces can be deleted while workspaces provided by a central configuration file [page 9] can not be deleted. To distinguish these two types, an read-only icon is attached to all entries that are not created by the user. The delete button will only be active if a user-created workspace is selected.

- *Hide*
  Check or uncheck this checkbox to hide or show a workspace in SAP Logon.
  To display all hidden Workspaces, select *View* > *Show All Workspaces* from the SAP Logon menu bar. Do the same to hide them again.
- *Name*
  Enter the corresponding column inside the table to change the name of a workspace.
- *Description*
  Enter the corresponding column inside the table to change the description of a workspace.

Hidden Entries

It is possible to hide selected system connections or folders in SAP Logon. To show or hide all entries, choose *View* > *Show Hidden Entries* or *View* > *Hide Hidden Entries* in SAP Logon menu. When the respective option is chosen, the hidden entries are shown again, colored in light grey. To finally display a hidden connection or folder again, simply right click on the corresponding entry and choose *Show Entry* in the context menu.

Filter

You can enter text to filter the list of connections. It checks if the filter text is contained in at least one of the fields *name*, *connection string*, *description* or *memo* of each connection. The list starts filtering when you press ENTER or automatically with a short delay after typing the last character in the filter entry field.

Context Menu

Use the context menu to organize connections or to choose view options.

- Connections
  - *Connect* to server
  - Create a *New* connection, *Edit* or *Delete* an existing connection
  - *New Link*: Creates a new connection entry as a link of an already existing system connection. The link takes over the connection string [page 53] of the original connection entry [page 9]. *Logon* [page 11] settings like the *Client*, *User* and *Language* data as well as custom *Security* [page 12] options can be set for each link individually independent of the original connection entry. To distinguish links from connections, an icon () is displayed next to the former one inside SAP Logon.
  - Create a *New Folder* (initial name is "New Folder")
  - *Move* connections or folders to another folder
- View options
  - *Show/Hide Memo*
  - *Show/Hide Columns* (only available with view option *List Columns*)
  - *Hide Entry*
    Hides a selected system connection or folder.

4.2 Preferences Dialog

You can change the configuration and options of SAP GUI for Java using the *Preferences* dialog .

To start the *Preferences* dialog from the SAP Logon or any SAP GUI for Java session [page 31],

- choose *File* > *Preferences* in the GUI menu
- or use keyboard command `Ctrl` + `Comma` (on macOS: `Cmd` + `Comma`).

On the left side of the *Preferences* window you can choose a Preferences category. The options of the chosen category are displayed on the right side. To adopt any changes in the options, press *Apply*.

4.2.1 Configuration

Logon & Proxies

See chapter Configure Connections for SAP Logon [page 9].

Trust Level Editor

The **Trust Level Editor** is a helpful tool to facilitate the management of custom trust levels. It contains the default trust levels of SAP GUI that work as an initial basis for the creation of user generated trust levels, stored in the file `SAPGUI.policy` [page 45]. For more details and information regarding the concept and implementation of trust levels see Chapter Concepts and Implementation [page 45].

The following functions can be executed via the dropdown buttons *Trust Levels* and *Permissions* on top of the textedit control or by invoking the contextmenu inside of it. Before selecting a button, mark a corresponding row inside the editor via the left mouse button to select a certain trust level, or invoke the contextmenu directly via the right mouse button.

- *Add Trust Level*: Invokes the *Trust Level Properties* dialog to add a custom trust level to the editor. The following fields are to be edited:
  - *Key*: Key name of a custom trust level "`CLevel<number>`". This value is not editable.
  - *Prinicipal Class*: From the dropdown list, choose `com.sap.platin.base.security.GuiSessionPrincipal` [page 47].
  - *Copy Permission From*: Select a default-, or a previously added trust level as a basis for creation.
  - *Display Name* & *Display Description*: These values are displayed on the *Trust level classification:<SID>* popup (*FILE* > *Edit Trust Level...*): "`<Display Name>: <Display Description>`".
  - *Trust Level Order*: This options affects the appearance of the list of available trust levels on the popup *Trust level classification:<SID>*. You can change the order using the mouse cursor.
  - *Assigned Systems*: Number of systems the trust level is assigned to. This information is stored and editable in the `trustClassification` [page 45] file.
- *Remove Trust Level*: Removes the selected trust level.
- *Trust Level Properties...*: Invokes the *Trust Level Properties* of an already defined trust level. Similiar result to the execution of *Add Trust Level*, except that the fields *Principal Class* and *Copy Permission From* are not to be edited and therefore not displayed.
- *Add Permission*: After the execution of this function, the entry `permission <PermissionClass>` is created inside the editor, right below the selected custom trust level. Finally, right click `<PermissionClass>` and select an entry from the contextmenu to add a permission. You may also have to enter a `"<target_name>"` or an `"<action>"`. For more information, see subchapter The Permission Entry [page 47].
- *Remove Permission*

4.2.2 Design

Themes

SAP GUI for the Java Environment can be displayed in different themes and schemes. To define a preferred *Standard Theme*, open the *Preferences* window and choose *Design* > *Themes*.

The following themes are available in SAP GUI for Java:

- Morning Horizon
- Evening Horizon
- Quartz Light
- Quartz Dark
- Quartz High Contrast Black
- Quartz High Contrast White

> **Note**
>
> The design of the window decorations (such as title bar, its buttons, and the borders) are specific to the window system. The customizing of these elements has to be done using the facilities of the window system.

The following Theme options are available:

- *Show themes for SAP Screen Personas only*

  If selected, only themes are displayed, which support SAP Screen Personas.
- Use multi-colored icons
  Use colored icons for Horizon themes.
- *Automatically adjust appearance*

  This option is only available for SAP GUI for Java on macOS, and only applicable to Quartz Light and Quartz Dark theme or Morning Horizon and Evening Horizon theme. If selected, the appearance of SAP GUI for Java follows the Light or Dark appearance of macOS.

The following *Focus Options* are available for focus customization:

- *Display focus as dashed rectangle*

  The focus is displayed with a dashed line drawing a rectangle round the focused object.
- *Display focus as corner brackets*

  The focus is displayed with a right-angular bracket in each corner of the focused object.
- *Animate focus movement in keyboard navigation*

  If selected, the transfer of the focus from one screen object to another is visualized with an animation.
- *Highlight focus with CTRL key*

  If selected, the focus is highlighted when pressing CTRL in case you want to locate it on the screen.

Colors

Scheme

> **Note**
>
> This option is available for Morning Horizon, Evening Horizon, Quartz Light, and Quartz Dark.

For the chosen *Theme*, a particular color scheme can be set. All possible schemes regarding the chosen theme are displayed in the list of schemes. To set a scheme, select one from the list and choose *Apply*.

Colors in System

Scheme

> **Note**
>
> This option is available for Morning Horizon, Evening Horizon, Quartz Light, and Quartz Dark. This option is only enabled when a session window with one of these themes is in front.

After connecting to an SAP System, a scheme can be set for all clients or for a specific client of this particular SAP System using the session window menu *File* > *System Scheme For <SYSTEMNAME>*. All possible schemes regarding the current theme are displayed in the list of schemes for the stated system and in the list of schemes for the stated client. To set a scheme for all clients of the system or for the specific client, select one from the respective list and choose *Apply*.

Sound Settings

Some activities like pressing a button, changing the state of a radio button or checkbox and switching tabs of a tabstrip can trigger an audio feedback by default.

Fonts

In this panel, you can change the font size using preset combinations. The following font sizes are available:

- *Use Small Fonts*
- *Use Medium Fonts*
- *Use Large Fonts*
- *Use Extra Large Fonts*

To check the effect of a changed font size, choose *Apply*.

Alternatively, you can set fonts and font sizes on an individual basis by choosing *Use Expert Configuration*.

Expert Configuration

Expert Configuration enables the possibility to customize different font settings for *Dialog Font* and *Fonts for Web AS ABAP*. Just switch to one of the tabs and click the button *Choose...* to change the corresponding settings.

> **Note**
>
> Not all fonts listed in the font chooser are able to display the complete character set. If a chosen physical font does not include the glyph of a character, a rectangular placeholder is displayed instead. To avoid placeholders for missing glyphs, choose logical fonts only. In the font chooser, logical fonts are shown in bold. The following fonts are logical fonts:
>
> - Dialog
> - DialogInput

- Monospaced
- SansSerif
- Serif

4.2.3 Desktop

This section contains information about the configuration of default applications and standard programs for SAP GUI for Java.

Applications

There are different *Desktop Intergration Models* available.

- depending on the operating system where SAP GUI for Java is installed on, or
- independent from the operating system, provided by the GUI, called SAPGUI. This mainly serves as an internal fall back.

You can change the *Desktop Integration Model* using the corresponding dropdown.

> **Note**
>
> Changing the Desktop Integration Model does not only effect the *Application* settings but also the *Mime Type* and *Printing* settings from the *Desktop* section of the *Preferences* dialog.

You can choose a default *Web Browser* and change the *Download Directory* as well as the *Upload Directory* of SAP GUI.

Enter an appropriate path into the corresponding editfield, or click on the button *Browse ...* to change the Web Browser or a Directory.

Mime Types

To handle MIME types, it is possible to set appropriate applications. You can also add additional extensions to a MIME type.

- Select a MIME type from the list and press the button *Add ...* to add an extension.
- Enter the path to an application into the corresponding editfield or simply press the button *Browse ...* to define an application to handle MIME types.

Printing

In this section, you can define a default queue for printing. Just select one element from the dropdown menu *Default Queue*.

> ⓘ Note
>
> To add a new queue to the dropdown, it has to be predefined in the operating system printer settings.

4.2.4 Other

Input History

The input by the user is stored locally by default.

> ⓘ Note
>
> Passwords typed in password fields are never stored in the history database.

To modify the Input History settings, the following options are available:

Setting	Function
Off	No input history is available.
On	Input history is available. Your input is stored locally in the database. When you enter data or press the backspace key, a list of recently entered data in the same field is displayed (ABAP field names must be available for this feature to work, which is not the case for "low speed connections", see The System Tab [page 11]). To disable or enable input history on a per field basis, open the context menu on the corresponding inputfield to see a menu item *Disable History* or *Enable History*.
Expires After: **X** *Month(s)*	Specifies when entries are considered obsolete and can be deleted.
Keep History for: **X** *Input Fields*	Specifies how many values are stored.
Display: **X** *Entries in History List*	Specifies how many entries will be displayed from the history list
Clear History	All history items are deleted. This applies across all systems.
Use mouse drag-down to open history list	If selected, you can open the history list with the left mouse button. To open the history list, click into an input field and keep mouse button pressed while dragging-down.
Drag distance	Determines the drag distance before the history list opens.

Accessibility

The following Accessibility features are available:

- **Accessibility Mode**
  - *On*: Enables the Accessibility Mode
  - *Off*: Disables the Accessibility Mode

  When the Accessibility Mode is enabled further controls are included in the tab chain (for example controls in read-only state).
- **Screen reader support**

  To make use of this feature, a third party screen reader has to be installed.

Embedding

It is possible to have multiple versions of SAP GUI for Java installed. You can set the default SAP GUI for Java version to use via *Embedding*.

- *Current default values*: Shows the current System Default as well as the User Default version of SAP GUI for Java
- *Installed Versions*: A list of all SAP GUI for Java versions that are istalled.
  - To set a default SAP GUI for Java version select one of the entries from *Installed Versions* and press the button *Set default GUI*. The changes will be displayed under *Current default values*.
  - To remove a default SAP GUI for Java version, press the corresponding button: *Remove default GUI*.
  - To adopt the changes press *Apply*.

Certificates

There are four categories of certificates:

- *Personal*: Contains all Personal certificates imported by the user.
  These certificates are used to validate the user by the server. They are stored in a user's folder. The folder should be empty after the first installation of SAP GUI for Java.
- *User Trusted*: Contains all certificates accepted or imported by the user.
  These certificates are used to validate servers via HTTP/HTTPS communication. The user imports or accepts them. They are stored in a user's folder. The folder should be empty after the first installation of SAP GUI for Java.
- *System Certificates*: Contains Certificate Authorities (CA) or Intermediate Root Certificates provided by the administrator.
  These certificates are stored in a system folder. For this category only the *Details* option is available.
- *JRE Certificates*: Contains a list of certificates inside the Java Runtime Environment and is displayed in read-only mode.
  If you choose this category, a list of certificates provided by JRE CA will be displayed. For this category only the *Details* option is available.

The following options are available for the *Personal* and the *User Trusted* certificates:

- *Import*: Imports a certificate from a local file.
- *Export*: Exports an accepted certificate to a file on the hard disk.
- *Remove*: Removes a certificate.
- *Details*: Displays a pop-up containing the data of the certificate.

> **Note**
>
> If your organization uses own Single-Sign-On certificates and a certificate chain, it is required to import the organization's CA certificates in the category *System Certificates*. Otherwise SAP GUI for Java is not able to verify the organization's servers by https connection, because the certificates provided by the servers do not have a CA certificate on the client side (inside JRE).

4.2.5 Web AS ABAP

General

Text Editing

- *Automatically move focus to next input field on reaching the end of input field*
- *Automatically select all text when tabbing into field (insert mode only)*
  After tabbing into an input field, the complete content is selected. This allows you to overwrite the text with the first character typed or to directly apply any clipboard actions with it.
- *Overwrite*
  Sets the input method to overwrite (OVR) at system start. Otherwise, the input method is set as insert (INS). You can switch between OVR and INS inside the system clicking on the corresponding entry on the right hand side of the status bar.

Clipboard

This option defines which text elements and input fields are copied to the clipboard using the corresponding function.

- *Copy text of entry fields and labels*
- *Copy text of enabled entry fields only*

Use `CTRL` + `Y` for tab delimited and `CTRL` + `SHIFT` + `Y` for space padded text.

Dragging

1. *Drag and Drop*

Screenshot

- *Fit to page*
  Enable this option so that the selected screen area fits the page. Otherwise, the actual size is printed.

SNC Encryption

- *Activate SNC encryption without SSO globally*
  You can connect to SAP systems via SNC encryption without a globally implemented single sign-on (SSO).
  For more information, see SNC Encryption [page 51].

HTML Viewer

Settings

- *Enable Developer Tools*
  Displays the *Inspect Elements* menu entry in the context menu for HTML views to open the *Develop Tools* window.
- *Open external links in popups*
- *Enable remote debugging port*
  Enter the port number.

Cache

- *Directory*
  Displays the directory for cached data.
  To clear the complete cache, choose *Clear*.

Source Editor

The Source Editor preferences allow you to configure the behavior of the built-in source code editor in SAP GUI for Java. These settings apply to Web AS ABAP connections and control how the editor is activated and how it behaves when editing ABAP source code.

Global Activation

- *Enable Source Editor*
  Activates the source editor for SAP GUI for Java.
  This checkbox acts as a global switch for the source editor feature across all systems:
  - Activated: The source editor is available system-wide. Users can activate it for each system in *ABAP Editor* (`SE38`) and *Object Navigator* (`SE80`) at *Utilities* > *Settings* > *ABAP Editor* > *Source code-based editor*.
  - Deactivated: The source editor is not available. The per-system option *Utilities* > *Settings* are not available.

> → Recommendation
>
> If you want to use the source editor on some systems but not on others, activate the global switch and control the behavior for each system in SE38 and SE80. Only disable the switch if you want to prevent the source editor from being used on all the systems.

> ⓘ Note
>
> Changes to this setting take effect only in newly opened connections.

Options

- *Highlight Current Code Block*
  Visually highlights the code block that currently contains the cursor.
  Activated: The source editor highlights the active code block (for example, an `IF` statement, a `LOOP`, or a `METHOD`) when you navigate through the source code. This helps you to identify the scope of the current context.
- *Automatically indent new lines*
  Automatically applies indentation when choosing ENTER to create a new line.
  Activated: New lines are indented to match the indentation level of the previous line. This helps maintain consistent formatting without requiring manual indentation after each line break.

SAP Screen Personas

SAP Screen Personas is available when it is active on the ABAP application server. The following settings affect all sessions of connections to ABAP application servers where SAP Screen Personas is active.

- Allow SAP Screen Personas to modify screens of the (local) SAP GUI for Java application. Switching on or off has an effect on new sessions only.
- Set the options for the cache, which stores resources like images.
- Use or don't use background images given by the flavor or the theme. Disabling this option displays the default background of the chosen design theme. This option is also available in the Personas menu of the SAP GUI for Java session.
- Use or don't use colors (such as font color, border color, or background color) given by the flavor or the theme. Disabling this option displays the default colors of the chosen design theme. This option is also available in the Personas menu of the SAP GUI for Java session.

Scripting

Scripting simplifies user interaction with SAP GUI by offering a macro-like interface that can be used for automating tasks. For more information, see Scripting [page 58].

Scripting

To enable or disable Scripting in SAP GUI for Java, select the corresponding combobox: *On* or *Off*.

Options

By the following options you can set notification messages about the scripting behavior and determine how a script has to be recorded:

- To enable notifications when *an external script calls into SAP GUI* or *an external script requests to open a connection*, the corresponding checkboxes have to be set.
- Recording can be done
  - *with absolute ID*

The connection number, session number, and window number are recorded by this option and must exactly correspond to replay a script.

```
application.findById("/app/con[<connection_number>]/ses[<session_number>]/
wnd[<window_number>]/...").<action>
```

- *with relative ID*
  This option is set by default. Only the window or user area are recorded and must correspond to replay a script.

  ```
  userarea.findById("...").<action>;
  ```

  ```
  window.findById("...").<action>;
  ```

- *with common SAP GUI ID*
  This option records a script by session and window number.

  ```
  session.findById("wnd[<window_number>]/...").<action>;
  ```

Scripts directories

The default directory where scripts are saved depends on the operating system that SAP GUI is installed on. To add additional directories or to remove existing directories, press the corresponding buttons *Add...* or *Remove*.

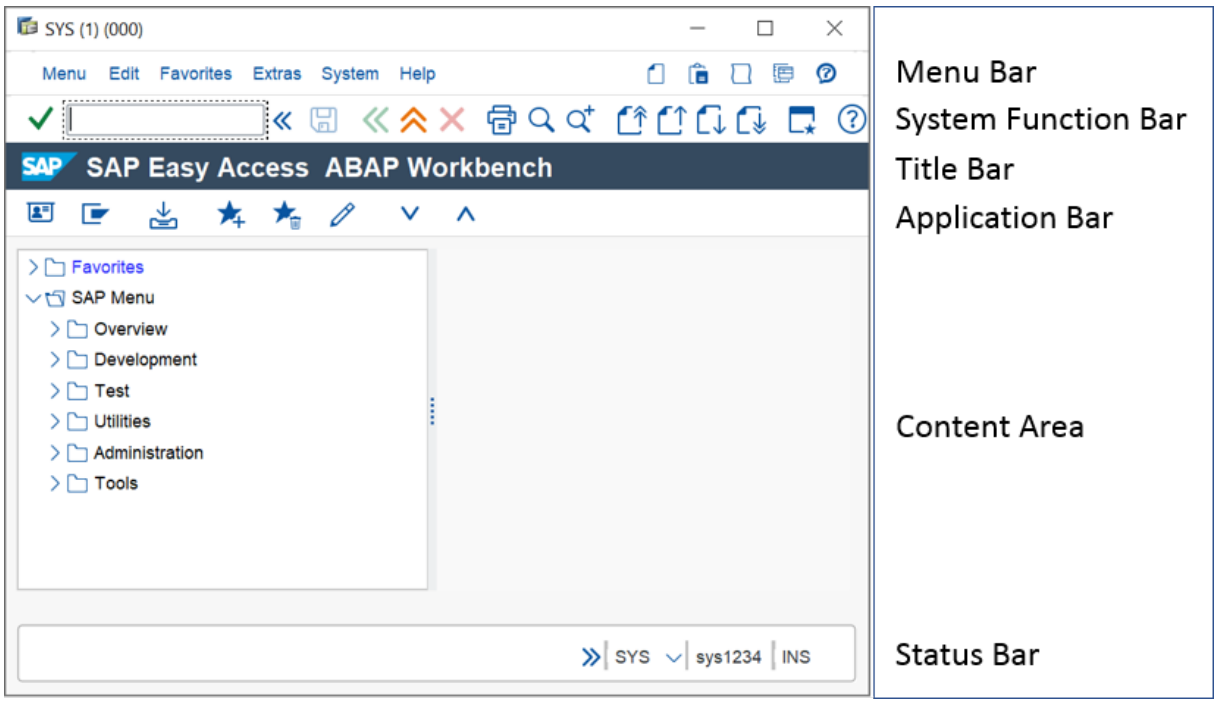
Additional Information

- **Status messages**
  - *Show status bar text in new dialog*: In addition to the status bar, a dialog displaying the status message is displayed. This can be enabled or disabled separately for each message type: *Success*, *Warning*, and *Error*.
- **Dropdown listbox**
  - *Always show keys*: In addition to the listbox text, keys are displayed for the corresponding item (if available).
    - *Sort items by key*: Instead of the text, listbox items are sorted by their key values if this option is enabled.
- **Window title**
  - *Show long window name*

Related Information

SNC Encryption [page 51]

4.3 Elements of an SAP GUI for Java Session Window



Menu Bar

The menu bar is the topmost bar on the screen. The menus on the left are provided by the ABAP program while the menus on the right are provided by the GUI itself.

> **Note**
>
> When SAP GUI for Java is running on macOS the menus provided by the GUI itself are included in the macOS menu bar.

The following menus are provided by the GUI:

Menu	Description
File	Contains some basic functions of session and security management as well as a link to the Preferences dialog.
Edit	Contains some basic editing functions like *Undo*, *Redo*, *Cut*, *Copy*, *Paste*, etc. and some options like *Accessibility Mode*
Only visible if SAP Screen Personas is enabled and available: *Personas*	Contains some functions of SAP Screen Personas. These functions are only enabled if SAP Screen Personas is active.

Menu	Description
Note In SAP GUI for Java on macOS the Personas menu is always visible.	
Scripts	Scripting Dialog. Access to scripting tools and the list of scripts located in the scripts directories to directly invoke a script.
Window	Contains some functions for window management. E. g. *Minimize*, *Zoom*, *Switch Window* and a list of all active sessions.
Help	Provides various forms of online help.

The following menus are available in every SAP application:

Menu	Description
System	This menu contains the functions that affect the whole system. For example, *Create session*, *User profile* and *Log off*.
Help	Contains access to the SAP Library, a Glossary, an application help and some other help functions.

The following menus are standard in most SAP applications:

Menu	Description
<Object>	Usually named after the object you are currently working with. It contains functions that affect the whole object. For example: *Display*, *Change*, *Print*, or *Exit*.
Edit	Allows you to edit components of the current object. Standard functions include *Select*, *Edit* and *Copy*. The *Cancel* option let you leave a task without saving the data you have entered.
Goto	Allows you to move directly to other screens of the current task. Also contains the *Back* option, which takes you back one level in the session hierarchy. Before going back, the system checks the data you have entered on the current screen and displays a dialog box if it detects a problem.

The following menus may also appear:

Menu	Description
Extras	Contains additional functions you can choose to complete the current object or an object component, but which you do not need regularly.

Menu	Description
Environment	Contains functions to display additional information about the current object.
View	This menu enables you to display the current object in different views. Example: Switching between a single-line and double-line display of a table.
Settings	Allows you to set user-specific transaction parameters.
Utilities	Allows you to do object-independent processing, such as deleting, copying, and printing functions.

System Function Bar

The system function bar is placed below the menu bar. It consists the Enter-Button, the command field with its open/close button, and buttons with frequently used functions. Following functions can be available in the system function bar:

- Save as Variant...
- Back
- Exit / Log off
- Cancel
- Print
- Find
- Find next
- First Page
- Previous Page
- Next Page
- Last Page
- Create Session
- Help

Title Bar

The title bar contains the name of the application currently being displayed.

Application Bar

The application bar is located below the title bar. The application bar contains buttons with functions relevant to the specific application.

Status Bar

The status bar provides general information on the SAP System and the transaction or task you are working on. System messages are displayed on the **left** side of the status bar.

There are three fields on the **right** hand side of the status bar: one containing server information and the other two with status information.

The status fields are described from left to right:

Status Field	Description
Message area	Displays system messages (success message, warning message, error message)
Open / Close button	Opens / closes the system information bar The following items will only be visible if the system information bar is opened.
Configurable session information	Display a session information which can be configured by the subsequent Menu button.
Menu button	Displays a menu where you can choose one of the following items to be displayed in the system information bar: • System • Client • User • Program • Transaction • only if SAP Screen Personas is available: Screen information
Application server field Example bsw6523	Displays the application server you are connected to
INS *or* OVR	Specifies your data entry mode. By clicking on this field (or by using the Insert key), you can switch between the insert mode (*INS*) and the overwrite mode (*OVR*).
only if SAP Screen Personas is enabled and available: SAP Screen Personas button	Indicates status information regarding SAP Screen Personas. Opens the Flavor Manager is SAP Screen Personas is active.

5 Reference

5.1 Installation Reference

You will find the necessary information here if you prefer to install SAP GUI for Java manually or remote as an administrator for many client PCs in parallel.

> **Note**
>
> See also chapter Installing or removing an SAP GUI for Java version [page 6] for general information how to install or to remove an SAP GUI for Java installation.

System Requirements

When using the installation package (**`PlatinGUI-<platform>-<version>.jar`**), you need an installed Java Virtual Machine (JVM). In this case, make sure you have installed a supported JVM and the necessary runtime libraries. For further information, see chapter Miscellaneous Product Information [page 63].

Installation Program

The complete installation program is contained in the SAP GUI for Java installation package **`PlatinGUI-<platform>-<version>.jar`**. The package is signed and versioned. You can verify the integrity of the package with the jarsigner program included with your Java Virtual Machine to make sure the code is authentic and the package is not damaged.

> **Code Syntax**
>
> ```
> jarsigner -verify -verbose PlatinGUI-<platform>-<version>.jar
> ```

You can find the full documentation for the jarsigner in JSE tool documentation from Oracle.

If the installation package detects that this version of the SAP GUI for Java is already installed it allows to remove or to reinstall the installation. On all platforms you can also use the command line installer to remove a installed version by calling

> **Code Syntax**
>
> ```
> java -jar PlatinGUI-<Platform>-<Version>.jar uninstall
> ```

On Windows platform you can remove a centrally-installed SAP GUI for Java using the Software control panel.

The installation package is executable and takes a command and options. Short options without parameters can be clustered. The general way to start the installation package is:

Code Syntax

```
java -jar PlatinGUI-<platform>-<version>.jar [command] [option]
```

Native installation programs:

- **Linux platform**

  Code Syntax

  ```
  PlatinGUI-Linux-Installation-<version> [command] [option]
  ```

  The installation commands and options are the same as the commands and options of the JAR package.
- **macOS platform**
  There is no native installation package for macOS, but a ready-to-copy application.
- **Windows platform**

  Code Syntax

  ```
  PlatinGUI-Windows-Installation-<version>.exe
  ```

Possible Commands

The following table lists the possible installation commands.

Command	Description
`install`	Installs SAP GUI, see installation options below, default if no command is specified.
`uninstall`	Removes installation.
`reinstall`	Reinstall already installed SAP GUI.
`version`	Output of detailed version information.
`installations`	Lists all installed SAP GUIs.
`validate`	Verifies the integrity of all installation files for an existing installation.

Installation Options

The following table lists the possible options for the `install` command.

Option	Description
`-? --help`	Display the installation options.

Option	Description
`-f --force`	Force installation and reinstall if already installed.
`-F --noforce` (default)	Don't force installation and don't reinstall, if already installed. **Note** Per default, the installer will do nothing if it detects that this version of the SAP GUI for Java is already installed.
`-v --verbose`	Provide verbose output on stderr
`-V --noverbose` (default)	Don't provide verbose output on stderr
`-s --standard`	Use standard options without user interaction. This is not a silent installation since progress indication and installation summary and all error dialogs are still displayed.
`-S --nostandard` (default)	Don't use standard options without user interaction. This is well suited for automatic graphical installations.
`-m --desktopicons` (default)	Install desktop menu and shortcuts.
`-M --nodesktopicons`	Don't install desktop menu and shortcuts.
`-g --gui` (default)	Use GUI for installation .
`-G --nogui`	Don't use GUI installation. **Note** If you turn off the GUI you will get error messages and a short installation summary on the console.
`-r --register`	Register the executable with system.
`-R --noregister` (default)	Don't register the executable with system.
`-d --installdir`	Specify the Installation directory.
`-a --modulePath`	Specify the module path to Java VM external modules.
`-i --inputfile`	Specify the installation response file
`-l --logfile`	Specify the file name for installation log.
`-t --trace`	Specify the trace keys separated with ":"

The `unistall` Command

Code Syntax

```
java -jar PlatinGUI-<platform>-<version>.jar uninstall
```

The **`uninstall`** command removes an installed version of the SAP GUI for Java.

The `version` Command

Code Syntax

```
java -jar PlatinGUI-<platform>-<version>.jar version
```

The version command just lists version information for the installation package, like the following example.

Output Code

```
Build date           : <year>-<month>-<day> 01:00:00 +0100
Build Info           : <host name>, <code line>, <build number>
Product type         : SAPGUI
Numeric Version      : A081000040100
Short Version Number : 8.10rev1
Long Version Number  : 8.10 rev 1
Product Dir Name     : <platform-specific folder name>
Short Version        : SAPGUI8.10rev1
Long Version         : SAP GUI 8.10 rev 1
Full Version         : SAP GUI for Java 8.10 rev 1
Official Version     : SAP GUI for the Java Environment 8.10 rev 1
```

External Installation Packages

You can choose to install optional external installation packages with the SAP GUI for Java.

Currently there is one well-known external package:

- **templates.jar**

This package is used to preconfigure the SAP GUI for Java, see Installation of Custom Data [page 42]. The installation program will look for all external packages in the same directory the installation program resides in.

Installation Response File

You can automate the installation process by providing an installation response file using the **`-i <responsefile>`** option to the install command.

Available Properties

The following table lists the available properties for the installation response file option.

Property	Description
installdir=<path> or **installpath=<path>**	Absolute path to installation directory. If this option is omitted a reasonable platform dependent default location is used (same as **-d <path>**).
disableui=true	Disables the interactive graphic installation program. All installation parameters have to be supplied using an installation response file or as command line options. All installation messages are written to standard output and standard error (same as **-G**).
automatic=true	If all necessary parameters are specified only show progress indication and error dialogs (same as **-s**).
reinstall=true	If this version of the SAP GUI for Java is already installed it is silently reinstalled and the installation proceeds as normal (same as **-f**).
uninstall=true	Remove the SAP GUI for Java installation if this version is installed, then exit (corresponds to the **uninstall** command).
logfile=<path>	Absolute path to installation log file including file name (same as **-l <path>**).
noshortcuts=true	Do not create desktop menus and desktop shortcuts (same as **-M**).
registerapplication=true	Centrally register executable application as application path on Windows, as link into /usr/bin on Linux (same as **-r**).

The file format is a list of lines with each one describing a property as a key value pair. Key names are delimited by " = " and " : ". These characters can be escaped in key names and in the value text using " \ " as escape character. Therefore a literal " \ " also has to be escaped.

Complete lines can be commented with "#".

Example of a Typical Response File:

Code Syntax

```
# Example of a typical installation response file
# suppress user interaction
disableui=true
# install to the default path
installidir=default
# reinstall gui if already present
reinstall=true
```

Installation Log File

Each installation of SAP GUI for Java writes a detailed protocol of all installation activities to the **sapgui.log** file. Please consult this file if you encounter any installation problems. If not changed, the default platform specific locations are:

- Windows and Linux platforms
  <home directory>/sapgui.log
- macOS platform (showing hidden files)
  <home directory>/Library/Logs/sapgui.log

5.2 Configuration Files

> **Note**
>
> This chapter contains SAP GUI for Java specific information regarding configuration files. For common information regarding SAP UI Landscape file, see SAP UI Landscape Configuration Guide.

Introduction to configuration files

SAP GUI for the Java environment stores all of its local configuration information and user preferences in files. The files are located in a place that is platform-specific:

- **Windows**:

  ```
  <home directory>\AppData\LocalLow\SAPGUI\
  ```

- **macOS**:

  ```
  <home directory>/Library/Preferences/SAP/
  ```

- **Linux**:

  ```
  <home directory>/.SAPGUI/
  ```

User specific settings for SAP GUI for Java

The following table lists files containing persisted information which can be accessed in *Preferences* dialog:

File name	Access path in Preferences	Description
globs	-	Associations of file name patterns to mime types
applications	*Desktop* > *MIME Types*	Document handler for specific document types (mime types)
externalCommands	*Desktop* > *Applications*	Well known applications: Browser, Editor, Upload Directory, Download Directory

File name	Access path in Preferences	Description
printSettings	*Desktop* > *Printing*	Print settings: Default spool queue, Queue Maps
settings	• Configuration • Design • Other • Web AS ABAP	SAP GUI accessible configuration variables
trustClassification	-	filled from TrustLevelPanel when connecting to a unknown back end, maps System IDs to TrustLevels The location of the configuration files is described in Configuration Files [page 46].
SAPGUI.policy	*Configuration* > *Trust Level Editor*	SAP GUI private Java security policy

Configuration files for connections and SAP GUI management

Configuration	File name	Description
Connection specific settings	SAPGUILandscape.xml	Connection documents
SAP GUI management	sapgui.registry	-

5.2.1 Centrally Managed Configurations

To allow for central configuration management within an enterprise, the local settings file (see local configuration file [page 42]) can include a reference to a central configuration file (which is typically located on an intranet web server). The format of local and central configuration files is identical.

> **Note**
>
> Locally stored settings always take precedence over settings taken from an included configuration file.

A configuration file can contain the following directives (they must occur only once for each configuration file):

> **Code Syntax**

```
@INCLUDE = "<included_configuration_file>"
@SAPUILANDSCAPE = "<path_to_(central)_Landscape.xml>"
@ProxyPACURL = "<path_to_ProxyPACURL>"
@proxyMode = "<proxyMode_value>"
```

Anything following a '#' on a line is ignored as a comment.

All information regarding message servers, routers, system descriptions, etc. as well as the custom connection entries are stored in one single SAP UI Landscape file. Used as a central SAP UI Landscape file, SAP Systems

relevant to all users can be put into in this file, so that they are available at all front-end computers. In such an environment, the only setup activity required by the end user is to open the Preferences [page 20] dialog and to enter the URL for the central configuration file (with an include parameter pointing to a SAP UI Landscape file) in *Configuration* > *Logon* > *Configuration File*, see also Configure Connections for SAP Logon [page 9].

5.2.2 Local Configuration Files

Information stored in the 'settings' file

- Location of an included central configuration file (filename or URL) providing settings and refering to a central Landscape.xml file
- Options the user has defined in the preferences dialog (like font size, etc.)

The information in the settings file is managed by the Preferences [page 20] dialog.

Information stored in the SAPGUILandscape.xml file

- Connections to SAP systems defined by the user
- Hierachical structure on the SAP logon items when using hierarchical view
- User specific notes attached to a connection string
- System descriptions
- Message server entries
- Router entries
- URLs for getting a system status of an SAP System

These local configuration files are created empty when the SAP GUI for Java is started the first time, or they import the values stored in `connections` and `connectionTree.XML` from versions before 7.40, if present. Please note that this import only happens once.

> **Note**
>
> The configuration files are read and written with UTF-8 encoding to ensure that characters not part of the current native encoding are correctly displayed and stored. Also, the configuration files used for centrally-managed configuration must be edited with an editor capable of storing in UTF-8 format.

5.2.3 Installation of Custom Data

The installation program for the SAP GUI for Java can include user provided data in the installation.

This is mostly used to preconfigure the SAP GUI for Java installation.

The custom data is transported in a preregistered optional, external installation package named **templates.jar**.

Possible use cases are the installation of configuration templates or the installation of custom certificates.

The SAP GUI for Java currently recognizes the following user supplied files:

- **settings.template**
- **trustClassification.template**
- **applications.template**
- **globs.template**
- **externalCommands.template**
- **security/truststore.ts** (make sure to create and add the security directory to the templates.jar)

Use of templates to preconfigure a SAP GUI for Java Installation

If a user starts up the SAP GUI for Java and has no personal configuration in place, his personal configuration files (settings, trustClassification, applications, globs and externalCommands) are created and prefilled with information found in the according *.template file.

If a configuration file already exists it is not altered.

Use of Custom Certificates Installed With the SAP GUI for Java

In addition to the user specific and the system specific trust store for server certificates the SAP GUI for Java also allows to store server certificates in a trust store inside the installation. This information is readonly after installation and resides in a standard java keystore in security/trustStore.ts

Example with templates for settings

Example

1. Create files named **`settings.template`** with your customized content.
2. Pack this file into a jar archive named templates.jar:

```
jar -cf ./templates.jar settings.template
```

3. Put the **`templates.jar`** file in the same directory as the SAP GUI for Java installation jar.
4. Start the usual installation process.

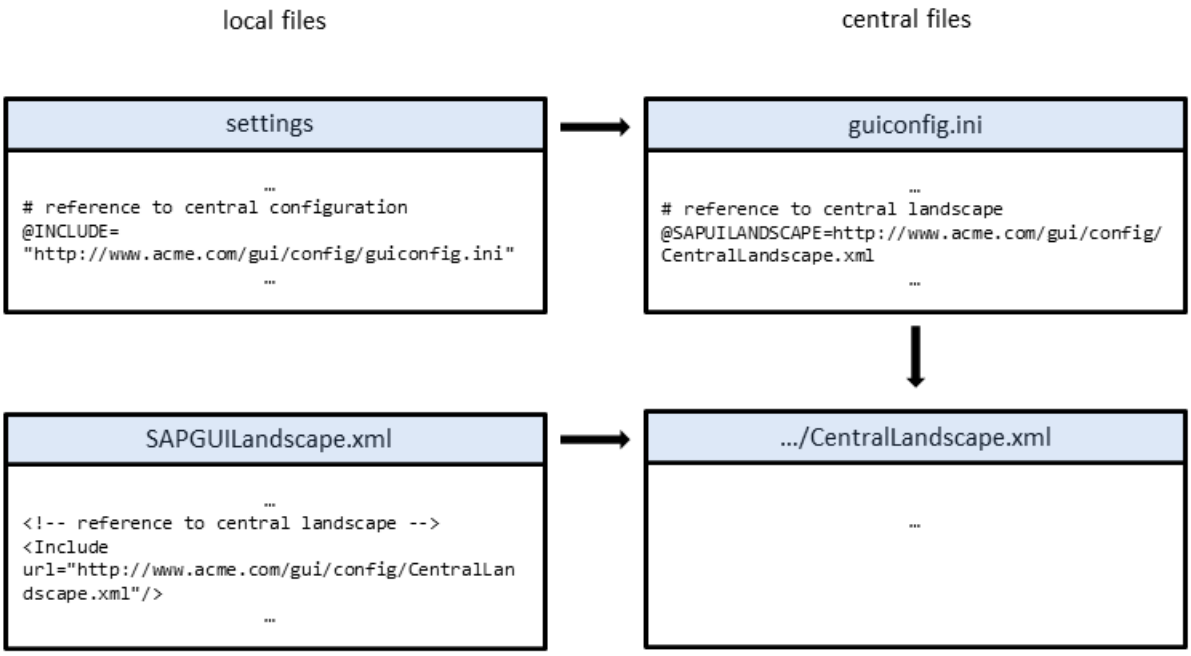
Note

These template files will only be activated, if the corresponding configuration files do not yet exist, to make sure changes previously made by the user on the local client are not accidentally overwritten.

5.2.4 Overview on Local/Central Files

The following graphic displays the references from

- the local settings to the central guiconfig.ini file
- the SAPGUILandscape.xml to the central Landscape.xml
- the central guiconfig.ini to the central Landscape.xml



5.3 Security Policy

Introduction

Every discussion about security always needs a backing security policy to define the meaning of security and to define what is acceptable and what is considered a threat.

The security concepts of the SAP GUIs are not concerned about securing the application server or to define who is allowed to connect to an application server but in contrary they deal with the question of controlling the access of back end ABAP code to local front end resources.

The SAP GUIs can access the local file system, are able to start child processes and also can access the configuration of the front end operating system by reading or writing configuration data or accessing system APIs.

For an ABAP program the SAP GUIs can act as service provider for printing documents or storing data in local files or retrieve information from the front end.

Since the SAP GUIs are run with usual user privileges, the possibilities to access or alter data are confined to the privileges granted to the current user account, still there is potential to disclosure or damage of user private information.

Different application servers may impose different trust relationships. A productive internal system running the local organizations human resources applications needs to be handled differently than a locally accessible system used for program development, where a lot of different people have access and also have the possibility to write their own (maybe just buggy or even malicious) code and where it is always possible to have a negative impact to local information, or a external system belonging to a customer or business partner.

On the other hand the SAP GUIs have more knowledge about the context of operations than the front end operating system and so can base their access control decisions on the trust relationship of the application server and the specific application running on this server.

The security concepts and the possibility of the SAP GUIs to control the access to local resources will not be sufficient for areas with a very high sensitivity, where the control of all data input and data output is mandatory. The only really reliable way to run client software in such an environment is to run the client in its own change root, or virtualized, sand box without access to the data on the host operating system at all.

5.3.1 Concepts and Implementation

The SAP GUI for Java has a large number of relations to the host operating system. To access features of the desktop it has to be able to access system specific interfaces like read access to the registry or the mime type database access on XDG compliant Unix systems.

Often the SAP GUI for Java will need to run system utilities to spool documents for printing or to open an appropriate document viewer to display documents.

The SAP GUI for Java also provides mostly file based utility services to the server side application. Most of these services are contained in the ABAP class CL_GUI_FRONTEND_SERVICES.

The SAP GUI for Java tries to confine the access to local resources into private compartments exclusively reserved for the SAP GUI for Java, and only accessible to the current session. The back end only knows about four distinct path names on the front end. These are the “temporary directory”, the “SAP work directory”, the upload directory [page 24] and the download directory [page 24] for the SAP GUI for Java. The temporary and the SAP work directory depend on the system and the current session, e. g. "`.../<Temp>/<User>/SAPGUI/<SID>.<ID_Key>`". The life time of data in the temporary directory is restricted to the life time of the session. The life time of data in the work directory is defined by the application creating the data unless it is sensed that the data is specifically transferred only to display it, then it is also deleted when the session is closed. Data in the upload and download directory belongs to the user and is not managed by the SAP GUI for Java.

The back end is allowed to write and read data in the work and temporary directory and to create new files in the download directory. It is also allowed to upload data from the upload directory. All other directories including the SAP GUI for Java configuration information although accessible for the SAP GUI for Java are forbidden to the back end by default.

Trust level classification

Usually most of the back ends will form clusters of similar trust relationships. There will be several systems in productive use one really has to trust and requiring access to local resources as well there will be development systems belonging to the local organization which should not need special access to local resources.

These different structures can be addressed with trust levels.

A trust level is a special principal granted a set of permissions. You can assign this trust level to an arbitrary number of systems. All systems assigned to a trust level will then share this same set of permissions.

Trust levels are completely independent and are not comparable. There are no inheritance relations between trust levels. The set of permissions comprising a trust level has to be complete and explicit. If you want trust levels to share some of the permissions specify them on both of the trust levels.

You specify the permissions for a trust level by writing a grant entry for the trust level:

```
grant principal com.sap.platin.base.security.GuiSessionPrincipal "#Level3" {
  permission com.sap.platin.base.security.ApplicationPermission "<<ALL
FILES>>","openDocument";
  permission com.sap.platin.base.security.ApplicationPermission "*", "openURL";
};
```

This example defines the permissions for a a trust level with the technical key “Level3” which allows to show documents of all types from all local locations on your hard disk, and also allows to open an external browser to all URLs.

The possible trust levels are stored in your settings and their technical keys always start with a hash sign (“#”) since the GuiSessionPrincipal class assumes all principals with an system name starting with a hash to be trust levels.

Configuration Files

```
<system preferences>/SAPGUI.policy
<user preferences>/SAPGUI.policy
<system preferences>/trustClassification
<user preferences>/trustClassification
<user preferences>/settings
```

Configuration

The SAP GUI for Java loads its policy information from several different locations. The first set of policy entries is read from the default Java VM locations for the system wide Java VM policy definitions and the user specific definition. Then the SAP GUI for Java loads the policy file in its system wide preferences directory, and at last the policy file in the users preferences directory.

These policy files will define sets of permissions granted to different trust levels.

The statements in all these policy files are merged. Since the policy is strictly a list of permitted actions there are no conflicts between different policy sources.

Each system can be assigned a to trust level. And these assignments are persisted in the `trustClassification` file.

The assignment of a system to a trust level is done from the SAP GUI's connection list. On editing a connection one can assign the system to a trust level using the *Trust Level* combo box on the *Security* tab (essentially assigning the application server systemID addressed in the connection to the trust level).

A list of active trust levels together with technical keys, descriptions and trust level names is stored in the `settings` file. The relevant key is `@activeTrustLevels` for example:

```
@activeTrustLevels = "Level4:Productive:Fully trusted. Needs access to local
    resources;Level3:Internal:Generally trusted, does not require extensive
privileges."
@defaultTrustLevel="Level 3"
```

These settings will just provide two levels `Level 4` and `Level 3` and specify that the default value for the selection in the combo box will be `Level 3`.

5.3.2 Policy Syntax

The policy syntax used for the system global and user private policy files is documented in Java SE Documentation[2] [page 51]. Additionally the SAP GUI for Java provides you with information about the system an application server belongs to, the transaction and the program name of the code requesting a front end related operation.

So one can restrict permission statements to a specific trust level, application server transaction or ABAP report.

The Grant Entry

Code being executed is always considered to come from a particular "`code source`" (represented by an object of type `CodeSource`). The code source contains the location (URL) where the code originated from. Code is also considered to be executed as a particular principal (represented by an object of type `Principal`), or group of principals.

Each Grant Entry[2] [page 51] includes one or more "`permission entries`" preceded by optional `codeBase` and `principal name/value` pairs that specify which code you want to grant the permissions. The basic format of a grant entry is the following:

```
grant codeBase "URL",
    principal principal_class_name "principal_name",
    principal principal_class_name "principal_name", … {
  permission permission_class_name "target_name","action";
  permission permission_class_name "target_name","action";
  …
};
```

All non-italicized items above must appear as is (although case doesn't matter and some are optional, as noted below). Italicized items represent variable values.

A grant entry must begin with the word grant.

The `Principal`, and `CodeBase` Fields

The `codeBase`, and `principal` values[2] [page 51] are optional, and the order of these fields does not matter.

A principal value specifies a `class_name/principal_name` pair which must be present within the executing thread's principal set. The principal set is associated with the executing code by way of a Subject.

The `principal_class_name` may be set to the wild card value `*`, which allows it to match any `Principal` class. In addition, the `principal_name` may also be set to the wild card value `*`, allowing it to match any `Principal` name. When setting the `principal_class_name` or `principal_name` to `*`, do not surround the `*` with quotes. Also, if you specify a wild card principal class, you must also specify a wild card principal name.

The principal field is optional in that, if it is omitted, it signifies “any principals”.

A `codeBase` value indicates the code source location; you grant the permission(s) to code from that location. An empty `codeBase` entry signifies “any code”; it doesn't matter where the code originates from.

> **Note**
>
> A codeBase value has to be a valid URI as specified in RFC 3986 . So the source location for code on a Windows system could look like:
>
> ```
> grant codeBase "file:///C:/somepath/api/" {
>     ...
> };
> ```

The exact meaning of a `codeBase` value depends on the characters at the end. A `codeBase` with a trailing `"/"` matches all class files (not JAR files) in the specified directory. A `codeBase` with a trailing `"/*"` matches all files (both class and JAR files) contained in that directory. A `codeBase` with a trailing `"/-"` matches all files (both class and JAR files) in the directory and recursively all files in sub directories contained in that directory.

GuiSessionPrincipal

Since the system ID, the transaction and program are known to the SAP GUI for Java it tags the requests with this context information. This context information is provided to the user in the form of a special principal.

There are two general forms to write a `GuiSessionPrincipal`:

```
grant principal com.sap.platin.base.security.GuiSessionPrincipal
      "SYS:trans:prog:sessionID" {
      ...
}
grant principal com.sap.platin.base.security.GuiSessionPrincipal "#TrustLevelKey"
    {
    ...
}
```

SYS denotes the system ID of the application server, `trans` the current transaction code, `prog` the actual name of the ABAP report. `sessionID` will be set to a unique ID for the session. The session ID will only be valid for the currently running session so this field is not useful in a policy file but is needed to isolate concurrent sessions.

All of the parts after SYS are optional or can be replaced with a `"*"` to match an arbitrary filed value.

When checking a GuiSessionPrincipal against a grant clause written in the second form (with a principal name like "*#TrustLevelKey*" the system part of the principal name is expanded to the trust level of the system currently connected to.

The Permission Entry

The basic form of a Permission Entry[2] [page 51] looks like:

```
permission permission_class_name "target_name", "action";
```

A permission entry must begin with the word `permission`. The word `permission_class_name` in the template above would actually be a specific permission type, such as `java.io.FilePermission` or `java.lang.RuntimePermission`.

The "*action*" is required for many permission types, such as `java.io.FilePermission` (where it specifies what type of file access is permitted). It is not required for categories such as `java.lang.RuntimePermission` where it is not necessary - one either has the permission specified by the "*target_name*" value following the `permission_class_name` or not.

Items that appear in a permission entry must appear in the specified order (`permission`, *permission_class_name*, "*target_name*", "*action*"). The entry is terminated with a semicolon.

> **Note**
>
> Case is unimportant for the identifiers (`permission`, `codeBase`, etc.) but is significant for the `permission_class_name` or for any string that is passed in as a value.

FilePermission

The general forms of a `java.io.FilePermission` look like:

```
permission java.io.FilePermission "${/}path${/}to${/}file", "read,write";
permission java.io.FilePermission "${/}path${/}to${/}*", "read,write";
permission java.io.FilePermission "${/}path${/}to${/}-", "read,write";
```

A `FilePermission` represents access to a file or directory. A `FilePermission` consists of a path name and a set of actions valid for that path name.

The valid actions for a `FilePermission` are `read`, `write`, `delete` and `execute` or any comma separated combination.

The path name is the path to the file or directory granted the specified actions. A path name that ends in `"${/}*"` (where `"${/}"` is expanded to the local path separator) indicates a directory and all the files contained in that directory. A path name that ends with `"${/}-"` indicates a directory and (recursively) all files in sub directories contained in that directory. A path name consisting of the special token `"<<ALL FILES>>"` matches **any** file.

A path name consisting of a single `"*"` indicates all the files in the current directory, while a path name consisting of a single `"-"` indicates all the files in directories below the current directory.

In the context of the SAP GUI for Java there is no value in working with relative path names, since there is no real control over the current directory.

When specifying a `java.io.FilePermission`, the "*target_name*" is a path name. Please always use the `${/}` pattern to specify the path separator. The policy parser will expand this pattern to the correct path separator for the system it is running on.

ApplicationPermission

A `FilePermission` is a very low level permission without semantic information. One can not conclude from a file permission why a file has to be accessed. If data is to be uploaded to the back end or if the file has to be read to be displayed in a document viewer, or if a file has to be read because it is a executable to be started.

It is helpful to have more abstract semantically defined permissions.

- **Opening documents**
  For opening documents the SAP GUI for Java provides the "`com.sap.platin.base.security.ApplicationPermission`" with an action "`openDocument>`"

  ```
  permission com.sap.platin.base.security.ApplicationPermission "${/}path${/}to$
  {/}file[mime type]", "openDocument";
  ```

  Only one permission is needed to be able to allow all actions necessary to display a given local document in the appropriate associated application. One can restrict the file path with the same syntax as for a `FilePermission` and additionally can restrict the file type with the additional `mime type` qualifier (possibly using a `"*"` wild card like `"text/*"`)
- **Opening URLs in an external browser**
  Opening URLs in an external browser can be controlled using the `"com.sap.platin.base.security.ApplicationPermission"` with the `"openURL"` action.

  ```
  permission com.sap.platin.base.security.ApplicationPermission "http://
  www.site.org", "openURL";
  ```

  Wild cards can be used to specify URL ranges for instance to match all URLs of one domain (`*.domain.org`).
- **Letting the back end start executables**
  Using the `"com.sap.platin.base.security.ApplicationPermission"` with the `"openApplication"` action is not much different to using a `FilePermission` with the action `"execute"` but using an `"ApplicationPermission"` only needs one permission which is checked early on to allow everything that is necessary to allow starting a specific application.

  ```
  permission com.sap.platin.base.security.ApplicationPermission "${/}path${/}to$
  {/}executable", "openApplication";
  ```

5.3.3 Default Trust Levels

The SAP GUI for Java is delivered with five default trust levels.

For these levels there are the following permissions:

```
grant principal com.sap.platin.base.security.GuiSessionPrincipal "#Level5" {
permission java.security.AllPermission "<<all permissions>>";
};
grant principal com.sap.platin.base.security.GuiSessionPrincipal "#Level4" {
permission java.lang.RuntimePermission "getenv.*";
permission com.sap.platin.base.security.ApplicationPermission "<<ALL
```

```
FILES>>", "openDocument";
permission com.sap.platin.base.security.ApplicationPermission "$
{InternalPath:F_GUILOGON}", "openApplication";
permission com.sap.platin.base.security.ApplicationPermission "*", "openURL";
};
grant principal com.sap.platin.base.security.GuiSessionPrincipal "#Level3" {
permission java.lang.RuntimePermission "getenv.*";
permission com.sap.platin.base.security.ApplicationPermission "<<ALL FILES>>",
"openDocument";
permission com.sap.platin.base.security.ApplicationPermission "*", "openURL";
};
grant principal com.sap.platin.base.security.GuiSessionPrincipal "#Level2" {
permission com.sap.platin.base.security.ApplicationPermission "<<ALL FILES>>",
"openDocument";
};
grant principal com.sap.platin.base.security.GuiSessionPrincipal "#Level1" {
};
```

5.3.4 Bibliography

[1] Java SE 8 Security Documentation

[2] Default Policy Implementation and Policy File Syntax

[3] Risk assessment of Java Permissions

[4] RFC 3986 Uniform Resource Identifier (URI): Generic Syntax

5.4 SNC Encryption

You can connect to SAP systems via SNC encryption without a globally implemented single sign-on (SSO).

Prerequisites

- You have not set up single sign-on, specifically neither SNC_LIB nor SNC_LIB_64 are defined in the environment.
- The required CA certificates and the required intermediate certificates for the validation of a server certificate (to manage the SNC handshake) are available and valid. On Windows they have to be stored in the certificate manager, on macOS they are stored in the **Keychain Access**. On Linux a truststore file */etc/SAPGUI/truststore.der* is required.
- You have set the flag *Activate SNC Encryption without SSO globally* in the Preferences Dialog [page 20] under *Web AS ABAP* > *General* > *SNC Encryption*, and the SAP GUI for Java has been restarted after the activation.

SAP GUI for Java checks these prerequisites very early during its startup so that the cryptography library can initialize an appropriate security context to encrypt the communication to the application server.

5.5 Certificates Management

In order to manage local SSL authority certificates which are used with the HTML control, consider the following information depending on your OS platform:

- Linux platform
  On Linux, the NSS Shared DB is used, which you can think of a native certificate store.
  For more information, see https://chromium.googlesource.com/chromium/src/+/master/docs/linux/cert_management.md .
- macOS platform
  On macOS, all certificate management for the HTML-Control is delegated to the native macOS **Keychain Access** application which manages certificates macOS wide. For more information regarding this application, see https://support.apple.com/guide/keychain-access/welcome/mac .
- Windows platform
  On Windows, certificate management is handled by the native OS certificate manager, available as Certmgr.msc from the **Run** box.

5.6 Accessibility

SAP GUI for Java offers various accessibility-related settings and options. Below you will find an overview of them. Use the respective link in the list to get more information.

> **Note**
>
> Most settings are done in the Preferences Dialog [page 20].

- You can switch on or off the Accessibility Mode. When the Accessibility Mode is enabled further controls are included in the tab chain (for example controls in read-only state).
- You can use control-specific and application-specific keyboard commands in SAP GUI for Java. Furthermore, you can check the menu and context menu for keyboard commands.
- Keyboard Focus Options (see Themes [page 21])
  Go to *Edit* > *Preferences* > *Design* > *Themes* > *Focus Options* to choose a dashed rectangle or as corner brackets as keyboard focus, or to enable keyboard focus animation.
- Fonts [page 23]
  Go to *Edit* > *Preferences* > *Design* > *Fonts* to choose between small, medium and large font size, or use an expert configuration to change the font type and size.
- Themes [page 21]
  Go to *Edit* > *Preferences* > *Design* > *Themes* to determine the theme of SAP GUI for Java. You can choose for example a Light or a Dark theme, a High Contrast Black or a High Contrast White theme, or another theme.
- Colors [page 22]

Go to *Edit* > *Preferences* > *Design* > *Colors* to change the color scheme regarding a chosen theme.

> **Note**
>
> This option is not available for the High Contrast themes.

- Colors in System [page 23]
  After connecting to an SAP System, a scheme can be set for all clients or for a specific client of this particular SAP System using the session window menu *File* > *System Scheme For <SYSTEMNAME>*.

  > **Note**
  >
  > This option is not available for the High Contrast themes.

- Sound Settings [page 23]
  Go to *Edit* > *Preferences* > *Design* > *Sounds* to activate audio feedback or to choose between different Sound Schemes.
- Additional Information [page 30]
  Go to *Edit* > *Preferences* > *Web AS ABAP* > *Additional Information* to enable additional options of information or to show status bar messages in a popup dialog window.

  > **Note**
  >
  > This setting is only related to Web AS ABAP connections.

- SAP Screen Personas [page 29]
  Go to *Edit* > *Preferences* > *Web AS ABAP* > *SAP Screen Personas* to enable or disable background images or flavor and theme specific colors when SAP Screen Personas is active and modifies screens.

5.7 Connection Strings

Connection string is a technical term used within SAP GUI for the Java Environment configuration. A connection string describes a connection address for a destination, for example, an SAP system application server, rather like an Internet URL describes a location for a web page.

Simple Connections Strings

In its simplest form, a connection string contains an IP address and a port number. This information is sufficient for SAP GUI for the Java Environment to open a direct TCP connection to a destination, for example, an application server. IP address and port number are marked with the prefixes `'/H/'` and `'/S/'` (for service). Note that the port number for an SAP application server is by convention 3200 plus the two-digit SAP system number.

Example

Example for a simple connection string with an application server's IP address (172.16.64.17) and port number (3200):

```
/H/172.16.64.17/S/3200
```

If your network environment supports DNS (Domain Name Services), a host name can be used instead of the IP address in all kinds of connection strings. (This requires a correct DNS configuration at the front-end computer, for example, using the /etc/hosts file).

Example

Example with an application server's host name (host.example.org) and port number (3200):

```
/H/host.example.org/S/3200
```

Simple connection strings need not be resolved by the GUI application. Resolution of host names and symbolic service names is done by the operating system's network layer.

SAP Routers

In a WAN (Wide Area Network) environment, SAP routers are used to make connections to remote SAP systems that cannot be reached with a direct TCP connection. Passwords may be used for each SAP router to control access.

In order to make a connection, the front-end computer is responsible for providing the complete route to the destination, possibly including a chain of several SAP routers. Path information is not provided by the routers. (Strictly speaking, an SAP router is actually better described as an application level proxy with password capabilities and strict source routing).

The address for each router is specified by a simple connection string (with the router's host name and port number), optionally followed **`'/P/'`** and the router password. The path from the current location to the destination is described by appending all router addresses together, followed by the address of the destination SAP system. Thus, a connection string with SAP routers generally has the form **`<router 1><router 2>.....<router n><destination>`**.

Example

Example with two routers (gate.remote.org, port 3299, and gate.example.org, port 3298), the first using a password (secret), for a connection to the application server host.example.org, port 3200):

```
 /H/gate.remote.org/S/3299/P/secret/H/gate.example.org/S/3298/H/
host.example.org/S/3200
 <---------- 1st router ---------><------ 2nd router ------><------
app_server ------>
```

Connection strings including SAP routers are passed to the GUI's communication layer and resolved step-by-step by the routers on the path. If host names and symbolic service names are used, each router must have access to correct network configuration information to resolve them.

Message Servers and Logon Groups

For load balancing purposes, application servers from an SAP system are usually configured in logon groups, where each group serves a particular kind of user. The application servers in each group are assigned to users using a 'least-heavily-loaded strategy'. This load balancing is done by message servers. Each SAP system has exactly one message server, which can be reached using TCP on a specific message server port.

Care should be taken that the application server's port number is not confused with the message server's port number. In small installations the message server's host name can often be identical to the host name of an application server, the port number is however always different. Symbolic service names for message servers by convention have the form **'sapms<SID>'**, where **<SID>** is the SAP system ID.

Message server and group information can be used to address an SAP system in a connection string. The address of the message server is specified as a combination of message server host name, message server port and group name. This information is marked with the prefixes **'/M/'** (message server host name), **'/s/'** (message server Port) and **'/G/'** (logon group).

Example

Example with message server (host name alrmain, port number 4253) and logon group (SPACE):

```
/M/alrmain/S/4253/G/SPACE
```

Connection strings with message servers are resolved by SAP GUI for the Java environment by contacting the message server and retrieving the (simple) connection string of an application server for the specified group. This requires network access to the message server at the time the address is resolved.

SAP router connection strings may be used in combination with message server connection strings simply by specifying the router address before the message server address. The router is then used for contacting the message server and the resolved application server.

Symbolic System Names

The most user-friendly form of connection string addresses an SAP system only by its symbolic name (by convention, the system ID) and the logon group name. This information is marked with the prefixes **'/R/'** (for the symbolic SAP system name) and **'/G/'** (for the logon group name).

Example

Example with SAP system (ALR) and logon group (SPACE):

```
/R/ALR/G/SPACE
```

Connection strings with symbolic system names are resolved by SAP GUI for the Java Environment by looking up the symbolic SAP system name in the Message Server List (a text file containing a mapping between symbolic system names and message server addresses) and replacing the **/R/** part of the connection string with the resulting message server address. The result is a complete message server connection string which is then further resolved as explained above.

Formal Syntax

For the technically interested reader, the following EBNF grammar formally describes the syntax of connection strings:

```
<connection string> := [<router prefix>]<local>
<router prefix> := <router>[<router(2)>...<router(n)>]
<router> := "/H/"<host>"/S/"<service>[("/P/"<password>) | ("/W/"<password>)]
<local> := <simple>|<message server>|<symbolic>
<simple> := "/H/"<host>"/S/"<service>
<messageserver> := "/M/"<host>"/S/"<service>"/G/"<group>
<symbolic> := "/R/"<system>"/G/"<group>
<host> := <hostname>|<ipaddr>
<hostname> := (any DNS hostname)
<ipaddr> := (any IP address, in dotted decimal form)
<service> := <servicename>|<port number>
<servicename> := (any IP service name)
<port number> := (any integral number <= 65535)
<group> := <string data>
<system> := <string data>
<password> := <string data>
<string data> := (any ASCII string not containing '/'  or '&')
```

5.8 Connection Parameters

Within SAP GUI for the Java Environment, connection parameters are used to define all attributes of an SAP connection. Aside from the SAP system address, connection parameters may also contain logon information and information about connection speed, codepages, security settings, and so on.

Connection parameters may contain multiple fields separated by ampersand characters (&). Each field has the form **<key>=<value>**. All fields are optional and may occur in any order, except for the conn field, which is mandatory and should come first. Unknown fields are silently ignored.

The following fields are defined:

Parameter	Description
conn	Connection String [page 53] of the SAP system.
clnt	SAP client to fill in on logon screen (for example, "001")
user	User name to fill in on logon screen (for example, "guest")
lang	Language to fill in on logon screen (for example, "EN")
tran	Transaction to start after logon (for example, "BIBS")
systemName	When specifying an application server you can provide the system ID of the R/3 system to allow contacting its message server. This allows you to turn off expert mode after data entry.

Parameter	Description
sncon	If set to "true", Secure Network connections (SNC) is used for this connection (provided that a suitable SNC provider is installed). To be valid this parameter requires sncname to be specified.
sncname	SNC name of the SAP system (for example, "p/secude:CN=example, O=organization, C=DE"). To be valid this parameter requires requires sncon to be specified.
sncqop	SNC quality of protocol, one of the following numbers: • 1 - Authentication • 2 - Integrity • 3 - Encryption • 9 - Maximum available
manualLogin	Do not use the automatic login feature of SNC. Require manual login.
cpg	SAP codepage number (for example, "8000" for Japanese Shift-JIS). The default codepage is denoted by "0"
wan	If set to "true", WAN optimisations (for low speed connections) are enabled
wp	Reserved for the mySAP.com workplace
ssot	Reserved for the use of single sign-on in the mySAP.com workplace
sso2	Reserved for the use of single sign-on in the mySAP.com workplace
rfcid	Reserved for dialog RFC usage

Example

Example for a connection to an SAP system with the connection string /H/host.example.org/S/3200:

```
conn=/H/host.example.org/S/3200
```

Example

Example for the same SAP system, with a start transaction (BIBS):

```
conn=/H/host.example.org/S/3200&tran=BIBS
```

Note

Note that for reasons of simplicity and downwards compatibility, the simplest form of connection parameters (containing only a conn parameter with a connection string) can also be specified without the leading **'conn='**, for example, in the form of a connection string. This practice is, however, not recommended with respect to possible future changes.

5.9 Scripting

The SAP GUI for the Java Environment comes with a built-in JavaScript engine and can be automated by executing scripts that emulate user interaction. These scripts can be created manually by editing the JavaScript code, or by recording user interaction. While a simple script may be used to present data on a SAP GUI screen, more complex scripts can easily automate whole transactions and even open connections to SAP systems automatically.

For detailed information about the JavaScript engine, see the SAP GUI Scripting API documentation.

Scripting Preferences

To get access to the Scripting, you need to activate it in the Preferences dialog. Open *Preferences* > *Web AS ABAP* > *Scripting*. On the Scripting panel you can also set the Scripting Options or the Scripts Directories. For more information about these settings, see the **Scripting** section in Web AS ABAP [page 27].

Access to Scripting

In the *Scripts* menu of the SAP Logon or the session window you have access to the Scripting window and to the list of scripts located in the scripts directories to directly invoke a script. In the *Scripting* window you can

- open a script (via the menu item *File* > *Open Script*)
- edit a script in the editor pane
- save a script (*File* > *Save Script*)
- record a script (*Scripts* > *Record Script*)
- play a script (*Scripts* > *Play Script*)
- destroy a script (*Scripts* > *Destroy Script*)

5.10 Trace Information

Trace information is used to investigate and report bugs. SAPGUI for the Java Environment employs a flexible, component-based trace model that allows you to produce trace information specifically for the components of interest, usually on request from SAP support. This background information is provided to assist in setting trace options.

Keep in mind that tracing seriously reduces system performance and might compromise site security because of sensitive information recorded during the trace session. Use trace mode only when necessary and delete trace files as soon as they are no longer required.

Trace mode is always activated for each individual process, for example, all SAPGUI for the Java Environment connections running in the same process share the same trace mode. When the process ends (all connections and logon window closed) tracing is automatically disabled.

Enabling Trace Mode

From SAP Logon or any SAP session window, choose *FILE* > *Trace...* to open the trace options dialog (see below).

The Trace Options Dialog

The trace options dialog contains the following fields:

- *Trace Components*:
  The *Trace Components* text area contains a list of trace keys and a short description for each trace key. Each trace key must occur on a separate line. Everything following a double-slash (`//`) on a line is regarded as a comment and ignored. In the table below you find a list of trace keys.

  > **Note**
  >
  > Some trace keys (such as `CON`, `SES`, `CALL`, and `EVENT`) are already active.

- **Output Format**:
  Determines the additional information output before each line of trace output
  - *Date/Time*:
    Timestamp Information
  - *Thread*:
    Name of the active thread
  - *Millisecond Delays*:
    Delay in milliseconds since last trace output
- **Dump**:
  - *Component Tree*:
    All GUI components in the object tree
  - *Automation Table*:
    All GUI automation objects
- **Logfile for error / traces**:
  Specify a trace file name here (use absolute filenames). Leaving this field empty disables the use of trace files. Changes to this field close any previously active trace files. Trace files are always opened in 'append' mode.
  For OS X, note that you must use slashes (`/`) as file name separators.
  To enable tracing for selected components, check the checkbox *Start Trace*. The path to the trace file is displayed in a pop-up after chosing *Apply* or *OK*. Chose *Show Log* to display the trace file.

  > **Note**
  >
  > Errors and dump information are also written if the checkbox *Start Trace* is not checked.

Press the *OK* or *Apply* button to save the new settings (*Apply* keeps the dialog open, while *OK* closes it). Press *Cancel* to discard all changes and close the dialog.

In the following table, you will find a list of possible trace keys and their short description.

Component	Short description
GUI Kernel	
CON	Connection
SES	Session
CALL	Automation Calls
EVENT	GUI Events
C_NET	Network Communication
NET	Network Communication
PAR	Parser
POLICY	Policy setup
Control Enabling	
C_CET	Control Enabling Technology
CET	Control Enabling Technology
DMGR	Data Manager
DP	Data Provider
DPTAB	Data Provider Table
DRV	Data Provider Driver
SRV	GuiServices
Start and Logon	
APPLET	Browser
LOGON	Logon Functionality
RFC	startup with RFC id
GUI Components	
GBT	Button
GBX	Box
GCB	Check Box
GCBX	Combo Box
GCO	Container
GCT	Combo Text Field
GFW	Window

GUI Components	
GLN	Line
GMB	MenuBar
GMI	MenuItem
GMN	Menu
GSC	Scroll Container
GSF	Symbol Field
GTBN	Toggle Button
GTC	Text Field
GUA	User Area
GUA_R	User Area Raster
GSC_R	Scroll Pane Raster
GSI_R	Simple Container Raster

Control Components	
AEC	Source Editor Control
CHART	Chart Control
CTM	Context Menu Control
GRD	Grid Control
HTML	HTML Control
HTMLDUMP	HTML Content Dump
PICT	Image Control
TBCL	ToolBar Control
TEC	Textedit Control
TREE	Tree Control

DIAG Parser	
C_AGI	DIAG Parser Interface
C_CON	Connection
C_SES	Session (externam mode)
C_AREA	Area Adopter (Containers)
C_ATOM	Atom Adaptor (Containers)
C_MENU	Menu Adaptor
C_TABL	Table Control Adaptor
C_TOOL	Toolbar Adaptor

DIAG Parser	
C_WIND	Window Adaptor
C_PAR	DIAG Parser
C_EVT	Event Handler

Enabling Trace Mode when Starting with a Direct Connection (Shortcut)

If SAP GUI for the Java Environment is started with a direct connection, there is no logon window for adjusting trace settings before logon. If it becomes necessary to generate trace information before the first session window appears (for example, because the logon process itself is to be traced), the following procedure is required to set trace keys from the command line (only recommended if the desired trace information cannot be produced using the logon window):

1. Modify the command line invoking the GUI by inserting the parameter **-t**, followed by a single space and the required trace keys separated by colons (**:**). This parameter must be the first command-line parameter after **com.sap.platin.Launcher**. The actual command line is platform-dependent.
2. Trace information is output to standard error (depending on the operating system, this can be redirected appropriately).
3. If trace information is additionally to be written into a trace file, the file name can be specified by inserting the parameter **-l**, followed by the name of the trace file. Use forward slashes (**/**) to separate directory names.

Example

Example for the trace keys LOGON and RFC, showing only the relevant part of the command line:

```
... com.sap.platin.Gui -n -t LOGON:RFC -o /H/...
```

Example

Example for the same trace keys, with output also redirected to the local file **/tmp/guitrace.txt**:

```
... com.sap.platin.Gui -n -t LOGON:RFC -l /tmp/guitrace.txt -o  /H/...
```

6 Appendix

6.1 Miscellaneous Product Information

System Requirements, Release Notes, Functional Scope and Known Issues

For more information regarding SAP GUI for Java 8.10 refer to the SAP Notes listed in the following table.

Topic	SAP Note Number	SAP Note Title
System Requirements	3485708	SAP GUI for Java 8.10 Requirements
Release Notes	3485691	SAP GUI for Java 8.10 Release Notes
Functional Scope	454939	SAP GUI for Java functional scope compared to SAP GUI for Windows
Known Issues	2989892	SAP GUI for Java known issues

Further SAP Notes

The table below lists the most popular notes related to SAP GUI for Java (application area **BC-FES-JAV**) for your convenience.

Topic	SAP Note Number	SAP Note Title
Technical Information	146505	SAP GUI for the Java environment (Platform Independent GUI)
Problem Solving	1617725	SAP GUI for Java FAQ and problem solving
	2194340	Central SAP UI Landscape for SAP GUI for Java
Notification Service	1014347	Notification service for news on SAP GUI for Java
Release Dates	1229666	Expected release dates for SAP GUI for Java revisions
Maintenance Strategy	147519	Maintenance strategy / deadlines for SAP GUI for Windows / SAP GUI for Java
Bug Reporting	326558	Description of an error in the SAP GUI for Java
	683960	Most common traces for SAP GUI for Java

You can find further SAP Notes related to SAP GUI for Java via the SAP Notes search link on the SAP Service Marketplace by entering application area **BC-FES-JAV**.

6.2 Further Information Sources

SAP GUI on SAP Community network (SCN)

On SCN, there is a community dedicated to various SAP GUI related areas, including SAP GUI for Java: https://community.sap.com/topics/gui. It includes the homepage of SAP GUI for Java, all the latest news about versions, download links and the platform availability matrix (PAM) are available here. There is also a discussion forum for users of SAP GUI for Windows, SAP GUI for Java or scripting SAP GUIs, including eCATT, where users can share their experiences and help each other with questions or problems. Development teams within SAP monitor the discussion forum, however it is not to be treated as a bug reporter replacement: only official SAP Support can work on issues reported in a customer message using the SAP Support Portal.

Email Notification Service

If you are interested in receiving a notification by email when a new software download is available for SAP GUI for Java, please follow the steps described in the SAP Note 1014347.

Important Disclaimers and Legal Information

Hyperlinks

Some links are classified by an icon and/or a mouseover text. These links provide additional information.
About the icons:

- Links with the icon  : You are entering a Web site that is not hosted by SAP. By using such links, you agree (unless expressly stated otherwise in your agreements with SAP) to this:
  - The content of the linked-to site is not SAP documentation. You may not infer any product claims against SAP based on this information.
  - SAP does not agree or disagree with the content on the linked-to site, nor does SAP warrant the availability and correctness. SAP shall not be liable for any damages caused by the use of such content unless damages have been caused by SAP's gross negligence or willful misconduct.
- Links with the icon : You are leaving the documentation for that particular SAP product or service and are entering an SAP-hosted Web site. By using such links, you agree that (unless expressly stated otherwise in your agreements with SAP) you may not infer any product claims against SAP based on this information.

Videos Hosted on External Platforms

Some videos may point to third-party video hosting platforms. SAP cannot guarantee the future availability of videos stored on these platforms. Furthermore, any advertisements or other content hosted on these platforms (for example, suggested videos or by navigating to other videos hosted on the same site), are not within the control or responsibility of SAP.

Beta and Other Experimental Features

Experimental features are not part of the officially delivered scope that SAP guarantees for future releases. This means that experimental features may be changed by SAP at any time for any reason without notice. Experimental features are not for productive use. You may not demonstrate, test, examine, evaluate or otherwise use the experimental features in a live operating environment or with data that has not been sufficiently backed up.
The purpose of experimental features is to get feedback early on, allowing customers and partners to influence the future product accordingly. By providing your feedback (e.g. in the SAP Community), you accept that intellectual property rights of the contributions or derivative works shall remain the exclusive property of SAP.

Example Code

Any software coding and/or code snippets are examples. They are not for productive use. The example code is only intended to better explain and visualize the syntax and phrasing rules. SAP does not warrant the correctness and completeness of the example code. SAP shall not be liable for errors or damages caused by the use of example code unless damages have been caused by SAP's gross negligence or willful misconduct.

Bias-Free Language

SAP supports a culture of diversity and inclusion. Whenever possible, we use unbiased language in our documentation to refer to people of all cultures, ethnicities, genders, and abilities.

www.sap.com/contactsap

© 2026 SAP SE or an SAP affiliate company. All rights reserved.

No part of this publication may be reproduced or transmitted in any form or for any purpose without the express permission of SAP SE or an SAP affiliate company. The information contained herein may be changed without prior notice.

Some software products marketed by SAP SE and its distributors contain proprietary software components of other software vendors. National product specifications may vary.

These materials are provided by SAP SE or an SAP affiliate company for informational purposes only, without representation or warranty of any kind, and SAP or its affiliated companies shall not be liable for errors or omissions with respect to the materials. The only warranties for SAP or SAP affiliate company products and services are those that are set forth in the express warranty statements accompanying such products and services, if any. Nothing herein should be construed as constituting an additional warranty.

SAP and other SAP products and services mentioned herein as well as their respective logos are trademarks or registered trademarks of SAP SE (or an SAP affiliate company) in Germany and other countries. All other product and service names mentioned are the trademarks of their respective companies.

Please see https://www.sap.com/about/legal/trademark.html for additional trademark information and notices.