SAP Fieldglass

API Reference | PUBLIC
SAP Fieldglass
2025-10-31

Business Analytics API

Technical Specifications Guide

SAP

Content

1 SAP Fieldglass Business Analytics API Specifications

Describes the functionality of the SAP Business Analytics API resource.

The SAP Fieldglass Business Analytics resource allows clients to retrieve analytical data that a user has set up in the SAP Fieldglass application.

1.1 API Workflow

Provides an overview of the SAP Fieldglass API workflow using the OAuth 2.0 protocol.

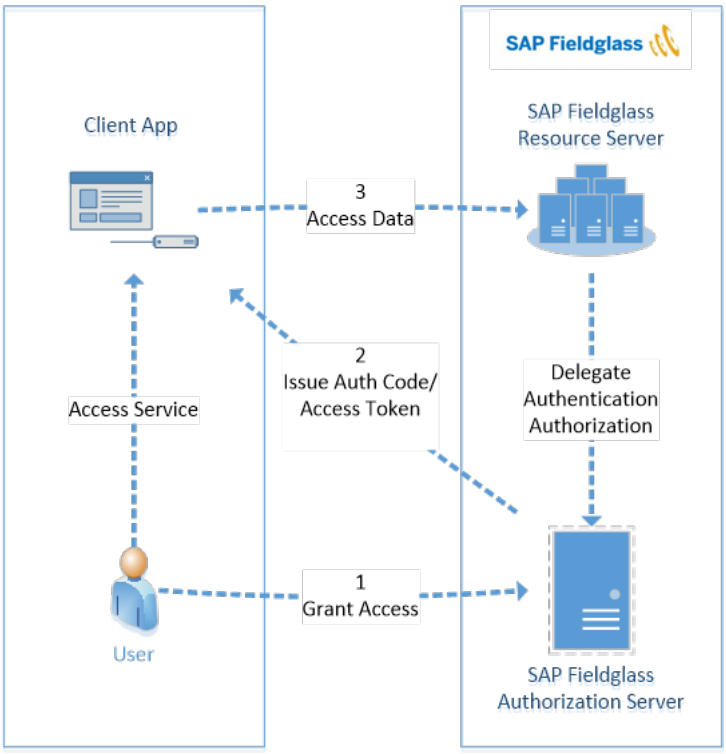
Workflow

The SAP Fieldglass REST API resource allows clients to send and receive integrated data directly against the application. The API uses the OAuth 2.0 protocol for authentication and authorization. SAP Fieldglass supports common OAuth 2.0 scenarios such as those for Web server, installed, and client-side applications.

To begin, you must obtain OAuth 2.0 client credentials - see Obtaining OAuth 2.0 Client Credentials. Next, your client application requests an access token from the SAP Fieldglass Authorization Server, extracts a token from the response, and then sends the token to the SAP Fieldglass REST API that you want to access.

All applications follow a basic pattern when accessing an SAP Fieldglass REST API using OAuth 2.0.

The following diagram illustrates the API authentication and authorization workflow.



1.2 Configuring Authentication and Authorization

The following steps summarize how to obtain OAuth 2.0 authorization and access token from SAP Fieldglass.

1. Obtain OAuth 2.0 client credentials - see OAuth2 Service Authentication.
2. Obtain an access token from the SAP Fieldglass Authorization Server.
3. Send the access token to an SAP Fieldglass REST API.
4. Refresh the access token, if necessary.

For details on how to perform each of these steps, see the following sections in this guide.

1.2.1 OAuth2 Service Authentication

There are three authentication options available for the OAuth2 service—client credentials, SAML authentication, and certificate-based authentication .

Client Credentials Authentication

Client credentials authentication provides a relatively simple mechanism for authentication, using a client ID and client secret. The client ID is a valid SAP Fieldglass username and the client secret can be either that user's

application password, or, to avoid issues with password rotation, a `<license key>`, as generated within the SAP Fieldglass system. Refer to the Create API Application Key or Web Service to create the license key.

The URL to use for client credentials requests is:

`https://<SAP Fieldglass Environment URL>/api/oauth2/v2.0?grant_type=client_credentials&response_type=token`

SAML Authentication

A SAML assertion can be generated for a valid user and passed via a 'SAML Response' parameter (note that this authentication still uses the client credentials grant type as a framework). Refer to Obtaining an Access Token from SAP Fieldglass Server [page 6] for more information

The URL to use for SAML authentication request is:

`https://<SAP Fieldglass environment URL>/api/oauth2/v2.0?grant_type=client_credentials&response_type=token`

Certificate Authentication

Certificate-based authentication provides a secure mechanism for authentication, eliminating the need for explicit credentials to be used (note that this authentication still uses the client credentials grant type as a framework). To leverage certificate authentication, do the following:

- Use a required X509 certificate issued by a trusted Certificate Authority (for example, DigiCert). Contact your SAP Fieldglass representative for more information.
- Use the correct URL.

  > **Note**
  >
  > The URL includes `auth` in the sub-domain.

  - For a test environment, the URL is in the following format: `https://<env_code>-auth.fgvms.com/ws2/api/oauth2/v2.0?grant_type=client_credentials&response_type=token`. For example: `https://xuat-fgvms.com/api/oauth2/v2.0?grant_type=client_credentials&response_type=token`.

    > **Note**
    >
    > Not all test environments are configured. You may need to submit a request to have it configured.

  - For U.S. production, the URL is: `https://auth.fieldglass.net/api/oauth2/v2.0?grant_type=client_credentials&response_type=token`.
  - For EU production, the URL is: `https://sso.fieldglass.eu/api/oauth2/v2.0?grant_type=client_credentials&response_type=token`.

To set up the X509 certificate for use in OAuth2 authentication, complete the steps outlined in the section within the SAP Fieldglass application. This adds the public certificate to the application and links it to a user (ensure it's a valid, active user), thus not requiring the credentials to be supplied as part of the OAuth2 request itself.

1.2.2 Obtaining an Access Token from SAP Fieldglass Server

Describes how to obtain an access token from the SAP Fieldglass server.

Before your client application can access private data using an SAP Fieldglass REST API, it must request an access token from the SAP Fieldglass Authorization Server to grant access to the API. A single access token can grant varying degrees of access to multiple APIs.

After an application obtains an access token, it sends the token to an SAP Fieldglass REST API in an HTTP authorization header or as part of the request body.

The x-ApplicationKey header is optional for Connector, Identity (SCIM), and Reporting APIs. If you receive an error for not including an API Key, you can obtain the API Key in your SAP Fieldglass Configuration Manager account.

Obtain Token via Username/Password Credentials

To obtain the access token using username/password credentials, send a `x-www-form-urlencoded` HTTP GET request specific to the username/password credentials of an SAP Fieldglass user.

The following sample code illustrates how to obtain the access token with user/password credentials.

Sample Code

sample request for token with user/password credential

```
POST /api/oauth2/v2.0/token?grant_type=client_credentials&response_type=token
                Authorization: Basic {Base64Encoded(user:credential)}
                Content-Type: application/x-www-form-urlencoded
                X-ApplicationKey: {server provided key}
```

Sample Code

sample response

```
{"access_token" : "WDXlKj3TTOn3rpg9GHnZpbKmvj1=",
                "token_type" : "Bearer","expires_in" : 7200}
```

Obtain Token via SAML Authentication

The access token can also be retrieved using SAML authentication. The primary role of SAML in online security is that it enables you to access multiple Web applications using one set of login credentials.

The SAML assertion is POSTed to the OAuth token endpoint, which in turn processes the assertion and issues an `access_token` based upon prior approval of the application. The client doesn't need a `client_secret` to be passed to the token endpoint.

The following sample code illustrates how to use SAML to obtain the access token with SAML.

Sample Code

sample request for token with SAML

```
POST   /api/oauth2/v2.0/token?
grant_type=client_credentials&amp;response_type=token&SAMLResponse={Base64Encoded
 SAML Assertion}
                Authorization: Basic {Base64Encoded(user:credential)}
```

```
                Content-Type: application/x-www-form-urlencoded
                X-ApplicationKey: {server provided key}
```

Sample Code

sample response

```
{"access_token" : "WDXlKj3TTOn3rpg9GHnZpbKmvj1=",
                "token_type" : "Bearer","expires_in" : 7200}
```

1.2.3 Token Expiration

Access tokens have limited lifetimes *(the current default is 7200 seconds = 120 minutes)*.

If your application requires access to an SAP Fieldglass REST API beyond the lifetime of a single access token, it needs to obtain a new token as outlined in Obtaining an Access Token from SAP Fieldglass Server [page 6].

If you make an API call using an invalid token, you receive a "401 Unauthorized" response back from the server. A token could be invalid and in need of regeneration for the following reasons:

- The token has expired.
- The user has revoked the permission initially granted to your application.
- You've changed the member permissions (scope) that your application is requesting.
- If a subsequent OAuth 2 flow has generated a new access token, then the previous token is invalidated.

A predictable expiry time isn't the only contributing factor to token invalidation. Be sure that your applications are coded to properly handle an encounter with a 401 error, by redirecting the user back to the start of the authorization workflow.

2 Report Data

Describes the API specifications for retrieving report details.

Base URL:`https://<SAP Fieldglass envionment URL>/api/vc`

Methods

HTTP Method	Action	URL
GET	Get All Reports [page 9]	`/report`
GET	Get Specified Report [page 11]	`/report/{report_id}`
GET	Get Report with Filtering [page 14]	`/report/{report_id}/{filter1_name}=[filter1_value1]`

Common Headers

Common Request Headers

Header	Required	Example Value	Data Type	Description
`authorization`	Yes	bearer *[access token]*	String	Security credentials for a login session.

Common Status and Error Codes

Code	Reason	Message	Description
200	Success response	OK	The request is valid and the response body has been created successfully. Refer to the specific method for details on a successful response body.
400	Error response	Bad request	The request is invalid. This response is returned if any of the following conditions are true: • The specified URL is invalid. • The request isn't formatted correctly. • The request is missing a required field.
401	Error response	Unauthorized	The email address and password used for authentication are wrong or expired.

Code	Reason	Message	Description
403	Error response	Forbidden or insufficient authorization	Token isn't verified.
404	Error response	Not found	The service being requested doesn't exist.
429	Error response	Too many requests	The user has sent to many requests in a given amount of time.
500	Error response	Unexpected system error	The request is valid and the response body encountered a system error. Contact your system administrator.

2.1 Get All Reports

Describes the API specifications for retrieving data for all reports.

Request

URL: `/api/vc/report`

HTTP Method: *GET*

Request Example

```
GET https://company.com/api/vc/report
authorization: Bearer WDXlKj3TTOn3rpg9GHnZpbKmvj1=
```

Response

Response Status and Error Codes

See Common Status and Error Codes [page 8] for more information.

Response Body

Parameter	Data Type	Description
`id`	String	The unique ID of the report.
`name`	String	The name of the report.
`metaURL`	String	The URL of the report's meta data.
`dataUrl`	String	The URL of the report's data.
`folderName`	String	The name of the folder the report is stored in.
`description`	String	A description of the report.

Parameter	Data Type	Description
properties	Object	An array of the fields contained in the report. Valid values: **Parameter** / **Data Type** / **Description** id / String / The property ID. name / String / The property name. dataType / String / The property's data type. maxLength / String / The maximum length of the property. analyticType / String / The property's analytic type. semanticType / String / The property's semantic type. measureUnitType / String / The property's measured unit type. measureUnits / String / The property's measured units. measureAggregation / String / Indicates if the measures are aggregated. scale / String / The property's scale.

Response Example

```
{
    "reports": [
    {
        "id": "z16100112470777028292937",
        "name": "API: Open Worker List",
        "metaUrl": "https://play.fgvms.com/api/vc/report/z16100112470777028292937/
metadata",
        "dataUrl": "https://play.fgvms.com/api/vc/report/z16100112470777028292937",
        "folderName": "My Folder",
        "description": "Open Worker List",
        "properties": [
    {
            "id": "WorkerID",
            "name": "Worder ID",
            "dataType": "STRING",
            "maxLength": 14,
            "analyticType": null,
            "semanticType": "ID",
            "measureUnitType": null,
            "measureUnits": null,
            "measureAggregation": null,
            "scale": 0
        },
        {
            "id": "Worker",
            "name": "Worder",
            "dataType": "STRING",
            "maxLength": 80,
            "analyticType": null,
            "semanticType": null,
```

```
            "measureUnitType": null,
            "measureUnits": null,
            "measureAggregation": null,
            "scale": 0
        }]
    },
    {
        "id": "z1610011247077702809878",
        "name": "API: Open Vendor List",
        "metaUrl": "https://play.fgvms.com/api/vc/report/z1610011247077702809878/
metadata",
        "dataUrl": "https://play.fgvms.com/api/vc/report/z1610011247077702809878",
        "folderName": "My Folder",
        "description": "Open Vendor List",
        "properties": [{
            "id": "VendorID",
            "name": "Vendor ID",
            "dataType": "STRING",
            "maxLength": 14,
            "analyticType": null,
            "semanticType": "ID",
            "measureUnitType": null,
            "measureUnits": null,
            "measureAggregation": null,
            "scale": 0
            },
        {
            "id": "Vendor",
            "name": "Vendor",
            "dataType": "STRING",
            "maxLength": 80,
            "analyticType": null,
            "semanticType": null,
            "measureUnitType": null,
            "measureUnits": null,
            "measureAggregation": null,
            "scale": 0
        }]
    }]
}
```


2.2 Get Specified Report

Describes the API specifications for retrieving data for a specific report.

Request

URL: `/api/vc/report/{report_id}`

HTTP Method: *GET*

Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
`report_id`	Yes	String	Unique ID of the report.	Path

Request Example

```
GET https://company.com/api/vc/report/z16100112470777028292937
authorization: Bearer WDXlKj3TTOn3rpg9GHnZpbKmvj1=
```

Response

Response Status and Error Codes

See Common Status and Error Codes [page 8] for more information.

Response Body

The output is dependent upon the report requested.

Response Example

```
"Time Sheet Status","Invoice ID","Site","Supplier","Time Sheet ID","Time Sheet
Submit Date","Invoice Type"
"Rejected",,"5.0","Anth","ANTETS00010360","07/14/2014 10:34 AM","Not yet Invoiced"
"Pending Approval (Level 2)",,"Bodo","Anth","ANTETS00007815","01/12/2013 12:17
AM","Not yet Invoiced"
"Pending Approval (Level 2)",,"Bodo","Anth","ANTETS00008394","03/30/2013 01:57
AM","Not yet Invoiced"
"Paid","ANTECD00000079","Add","Anth","ANTETS00007901","12/11/2012 02:53 AM","Credit/
Debit Memo"
"Paid","ANTECD00000080","Add","Anth","ANTETS00007901","12/11/2012 02:53 AM","Credit/
Debit Memo"
"Invoiced","ANTECD00000096","Add","Anth","ANTETS00007973","12/12/2012 01:43
AM","Credit/Debit Memo"
"Invoiced","ANTECD00000097","Bill","Anth","ANTETS00009891","02/16/2014 06:16
PM","Credit/Debit Memo"
"Approved",,"5.0","Anth","ANTETS00010378","07/14/2014 10:09 AM","Not yet Invoiced"
"Approved",,"5.0","Anth","ANTETS00010396","07/14/2014 10:07 AM","Not yet Invoiced"
"Approved",,"5.0","Anth","ANTETS00010432","06/24/2014 11:08 AM","Not yet Invoiced"
"Approved",,"5.0","Anth","ANTETS00010450","06/23/2014 03:08 PM","Not yet Invoiced"
"Approved",,"5.0","Anth","ANTETS00013456","03/12/2016 10:52 AM","Not yet Invoiced"
```

2.3 Get Reports in Async Mode

Describes the API specifications for retrieving data for reports in the async mode.

GET Execute Report in Async Mode

By setting the parameter `Asynchronous: true`, a report can be set in the async mode.

Request Format

URL: `<env_url>/api/vc/report`

HTTP Method: *GET*

```
curl --location '<env_url>/api/vc/report/<report_id>' \
--header 'isAsynchronous: true' \
--header 'Authorization: Bearer <token>'
```

Response Example

```
<?xml version="1.0" encoding="UTF-8"?>
<Status>
  <TransactionID>z2505201439438173037fcb</TransactionID>
</Status>
```

Other Responses or Errors

```
<?xml version="1.0" encoding="UTF-8"?>
<Status>
  <TransactionID>z250520205823152296711104</TransactionID>
  <ReturnCode>255</ReturnCode>
  <Message>Your transaction is already in progress. Please poll using the
transaction ID: [z250520205817053107871104] for the report output.</Message>
</Status>
```

GET Poll Report Transaction for Report Output data

Using the `transactionId` parameter, the status of a report can be viewed.

Request Format

URL: `<env_url>/api/vc/report/<report_id>?transactionId=<transaction_id>`

HTTP Method: *GET*

```
curl --location '<env_url>/api/vc/report/<report_id>?transactionId=<transaction_id>' \
--header 'isAsynchronous: true' \
--header 'Authorization: Bearer <token>'
```

Response Example

```
"Job Posting ID","Job Posting Status","Job Posting Create Date"
"CRPAJP00000501","Submitted","02/20/2012 05:32 pm"
"CRPAJP00000502","Submitted","02/20/2012 06:06 pm"
"CRPAJP00000503","Submitted","02/20/2012 06:36 pm"
"CRPAJP00000504","Submitted","02/20/2012 08:47 pm"
"CRPAJP00000505","Submitted","02/20/2012 08:58 pm"
"CRPAJP00000506","Submitted","02/20/2012 09:06 pm"
"CRPAJP00000507","Submitted","02/20/2012 09:11 pm"
"CRPAJP00000512","Submitted","02/20/2012 10:35 pm"
"CRPAJP00000513","Submitted","02/21/2012 12:05 am"
```

Other Responses or Errors

```
{"status":"Your transaction might still be in progress, as the report generation
typically depends on your configuration. Please wait and try polling again."}
{"status":"Your transaction might be an older or completed one. Please poll using
the latest transaction ID received from the asynchronous call."}
<?xml version="1.0" encoding="UTF-8"?>
<Status>
  <TransactionID>z2505202100069273025710 4</TransactionID>
  <ReturnCode>255</ReturnCode>
  <Message>Your transaction might be invalid. Please use the latest transaction ID
received from the asynchronous call to poll.</Message>
</Status>
```

2.4 Get Report with Filtering

Describes the API specifications for retrieving report data using one or more filters.

> **Note**
>
> In order to use a filter, it must be added as a filter in the report setup. Any filters configured on a report take on the value set there unless overridden at the API level.

Multiple filters can be applied in a single query. To properly construct a filter, do the following:

- Separate multiple filter values by the pipe (|) symbol. For example: `filter1_name=filter1_value1|filter1_value2`
- For a date/time filter, the first value is the start date/time and the second value is the end date/time. If no second date/time filter is provided, the value defaults to the current date/time.
- Separate multiple filters using an additional forward slash (/) followed by the filter name and value. For example: `filter1_name=filter1_value1/filter2_name=filter2_value1`

Request

URL: `/api/vc/report/{report_id}/{filter1_name}=[filter1_value1]|[filter1_value2]/{filter2_name}={filter2_value1}|{filter2_value2}..`

HTTP Method: *GET*

Request Parameters

Parameter	Required	Data Type	Description	Parameter Type
`report_id`	Yes	String	Unique ID of the report.	Path
`filter_name`	Yes, at least one filter name must be included.	String	The name of the filter.	Query string
`filter_value`	Yes, at least one filter value must be included.	String	The value of the filter.	Query string

Request Example

```
GET https://company.com/api/vc/report/z14011711012989849412906/Edited On=2020-05-01
00:00:00.0|2020-05-30 00:00:00.0
authorization: Bearer WDXlKj3TTOn3rpg9GHnZpbKmvj1=
```

Response

Response Status and Error Codes

See Common Status and Error Codes [page 8] for more information.

Response Body

The output is dependent upon the report and filter requested.

Response Example

```
"Time Sheet ID","Job Posting ID","Time Sheet Submit Date","Time
Sheet Approved Date","Time Sheet Status","Invoice ID","Main
Document ID","Site","Non-taxable","General Ledger Account Code","Invoice
Type","Worker","Business Unit","Task Name","Task Code","Supplier","Can
Invoice?","Cost Center","Currency","Holiday?","Main Document Type","General
Ledger Account","Main Document Owner","Main Document ID","Site","Non-
taxable","General Ledger Account Code","Invoice Type","Worker","Business
Unit","Time Sheet Status","Task Name","Task Code","Supplier","Can Invoice?","Cost
Center","Currency","Invoice ID","Holiday?","Main Document Type","General Ledger
Account","Main Document Owner"
"ANTETS00022612","ANTEJP00003075","05/23/2020 10:15 PM",,"Pending Approval
(Level 1)",,"ANTEJP00003075","Add",,"GLA1","Not yet Invoiced","SRD_PostDeploy
[a-zA-Z][a-zA-Z][0-9][0-9][0-9], SRD_PostDeploy","Anthem","Outsourced
Services","S001","Ayah","Yes","Prue.Creator","USD","No","Job
Posting","GLA1","Anthem1","ANTEJP00003075","Add",,"GLA1","Not yet
Invoiced","SRD_PostDeploy [a-zA-Z][a-zA-Z][0-9][0-9][0-9],
SRD_PostDeploy","Anthem","Pending Approval (Level
1)","Outsourced Services","S001","Ayah","Yes","Prue.Creator","USD",,"No","Job
Posting","GLA1","Anthem1"
"ANTETS00022597","ANTEJP00000782","05/16/2020 08:19 AM",,"Pending Approval
(Level 1)",,"ANTEJP00000782","Prue.Creator BU",,"GL Account - Dan","Not yet
Invoiced","A, Ayah-3.15.10","Prue.Creator BU 1 (default)","TskCode 1","TskCode
1","Ayah","Yes","Prue.Creator","USD","No","Job Posting","General Ledger Account -
Dan","A, Prue.Admin","ANTEJP00000782","Prue.Creator BU",,"GL Account - Dan","Not
yet Invoiced","A, Ayah-3.15.10","Prue.Creator BU 1 (default)","Pending Approval
(Level 1)","TskCode 1","TskCode 1","Ayah","Yes","Prue.Creator","USD",,"No","Job
Posting","General Ledger Account - Dan","A, Prue.Admin"
```

```
"ANTETS00022595","ANTEJP00003075","05/21/2020 12:12
AM",,"Rejected",,"ANTEJP00003075","Add",,"GLA1","Not yet Invoiced","SRD_PostDeploy
[a-zA-Z][a-zA-Z][0-9][0-9][0-9], SRD_PostDeploy","Anthem","Outsourced
Services","S001","Ayah","Yes","Prue.Creator","USD","No","Job
Posting","GLA1","Anthem1","ANTEJP00003075","Add",,"GLA1","Not
yet Invoiced","SRD_PostDeploy [a-zA-Z][a-zA-Z][0-9][0-9]
[0-9], SRD_PostDeploy","Anthem","Rejected","Outsourced
Services","S001","Ayah","Yes","Prue.Creator","USD",,"No","Job
Posting","GLA1","Anthem1"
```

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

Material Number: 20190313