Oracle Financial Services

REST APIs for Accounting Foundation Cloud Service

Release 22D
F76058-01
January 2023

ORACLE

Oracle Financial Services REST APIs for Accounting Foundation Cloud Service, Release 22D

F76058-01

Contents

List of Tables

1
Get Help

This section describes the Get Help options in AFCS Core Functions 22D Release.

1.1 Get Help in the Service

Use the Help icon

to access help in the application. If you do not see any help icons on your page, click your user image or name in the global header and select Show Help Icons.

Note that not all pages have help icons. You can also access the Oracle Help Center to find guides and videos.

1.1.1 Watch Video

Watch: This video tutorial shows you how to find and use help.

You can also read about it instead.

1.1.2 Additional Resources

- **Community:** Use Oracle Cloud Customer Connect to get information from experts at Oracle, the partner community, and other users.
- **Training:** Take courses on Oracle Cloud from Oracle University.

1.2 Learn About Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program. Videos included in this guide are provided as a media alternative for text-based topics also available in this guide.

1.3 Get Support

You can get support at My Oracle Support.

For accessible support, visit Oracle Accessibility Learning and Support.

1.4 Get Training

Increase your knowledge of Oracle Cloud by taking courses at the Oracle University.

1.5 Join Our Community

Use Cloud Customer Connect to get information from industry experts at Oracle and in the partner community. You can join forums to connect with other customers, post questions, and watch events.

1.6 Share Your Feedback

We welcome your feedback about Oracle Applications user assistance. If you need clarification, find an error, or just want to tell us what you found helpful, we'd like to hear from you.

You can email your feedback to My Oracle Support.

Thanks for helping us improve our user assistance!

2

About the REST APIs

You can use Oracle REST APIs to view data stored in the Oracle Financial Services Accounting Foundation Cloud Service. A REST API (also known as RESTful API) is an Application Programming Interface (API or web API) that conforms to the constraints of REST Architectural Style and allows for interaction with RESTful Web Services.

AFCS provides Representational State Transfer (REST) APIs to perform user tasks in an automated manner. The REST API is an Application-Programming Interface that provides a simplified way to exchange data through HTTP requests from a Client to the Server. In REST APIs, a resource is an object with a type, associated with data, and relationships formed to other resources. You can use a set of HTTP Methods to access each resource.

Resources are organized in a Hierarchical Structure that enables:

- Better organization, by grouping related data so that you can efficiently customize the resources.
- Improved performance by using a single HTTP request to handle multiple resources.

3
Getting Started

3.1 Prerequisites

- Access to the Accounting Foundation Cloud Service.
- Appropriate User Privileges to access the services.
- Technical and Functional knowledge to understand and execute the REST APIs and configuration knowledge.
- Knowledge of REST Concepts, JSON, and browser-based REST Client.
- Knowledge of interactive and automatic tools to verify the APIs such as Postman and Command Line Interfaces (CLI).

3.1.1 Obtain Account Information

The account creation email from Oracle contains the Identity Domain Name for the Accounting Foundation Cloud Service Instance. If you do not have this information, then contact your Service Administrator.

3.2 Authentication

The Authentication Process involves the use of cURL Commands in a CLI Tool to generate the access token and invoke REST APIs. The Authentication Token is generated through the OAuth Client ID and Secret Credentials created in IAM/IDCS during Provisioning. The Authentication Token does not require that you log in to the AFCS Application to invoke the REST APIs from external applications.

Ensure that you have the appropriate log-in credentials to access the Accounting Foundation Cloud Service and the appropriate roles to perform specific operations using the API Resources. The following is the list of steps for Authentication and further subsections provide the details:

1. Download the Application Certificate
2. Get the OAuth Client ID and Client Secret
3. Generate the Access Token
4. Invoke the API using the Access Token

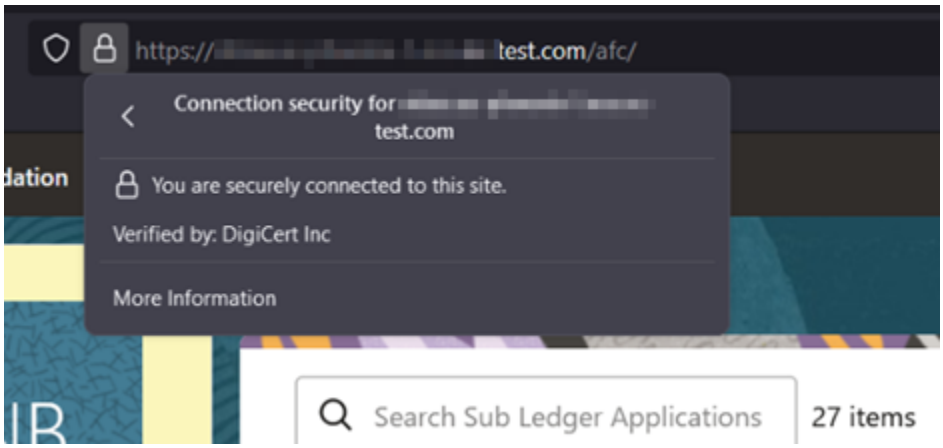
3.2.1 Download the Application Certificate

The Application Certificate is required for verification purposes when you use cURL commands. You may choose not to download the certificate if you plan to turn off the cURL Certificate Verification and use an insecure connection (if you add the --insecure Flag to the cURL command).

To download the Application Certificate, do as follows:

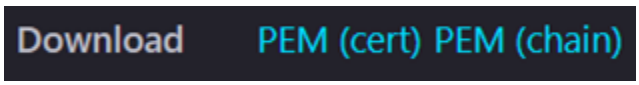
1. Log in to the AFCS Application.
2. Click the **View site information/Verified by**  Icon in the Browser URL Address Bar.
3. Select **More information**.

Figure 3-1 Certificates - More information



4. Click **View Certificate** and then click **PEM(cert)** to download the certificate.

Figure 3-2 Download PEM (cert)



3.2.2 Get the OAuth Client ID and Client Secret

To get the OAuth Client ID and Client Secret, follow these steps:

1. Enter the IDCS URL in the Browser's URL Address Bar.

   The **Oracle Cloud Account Sign In** Window appears.
2. Log in to **Oracle Identity Cloud Service** (IDCS).

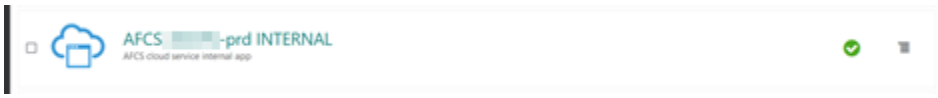
3. Select the Navigation Icon to view a list of available functions.

4. Select **Oracle Cloud Services**.

   For more information, see Access Service Consoles from **Administering Oracle Identity Cloud Service**.

5. From the Oracle Cloud Services Window, select the required AFCS Internal Application Service (in **AFCS <tenant-id> INTERNAL** Format) from the list.

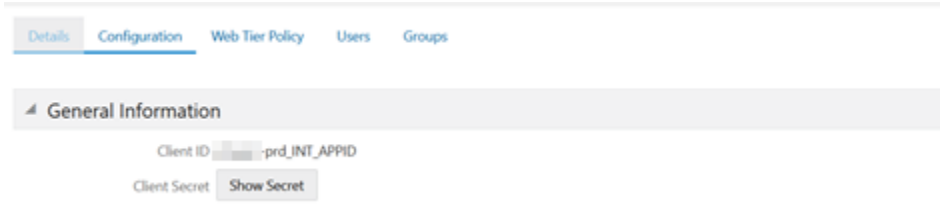
Figure 3-3 AFCS <tenant_id> INTERNAL



6. Click the **Configuration** Tab.

   The Client ID and Client Secret Details are displayed in the General Information Section.

Figure 3-4 Client ID and Client Secret



7. Copy the **Client ID** and **Client Secret**.
8. Open a CLI Tool.
9. Generate the Access Token as shown in the following section.

3.2.3 Generate the Access Token

To generate the Access Token, add the Client ID, Client Secret, User Name, and Password using cURL Commands in the CLI Tool. The following is an example:

```
echo -n "<tenant_oauth_app_client_id>:<tenant_oauth_app_client_secret>" |
base64 -w 0      (generates basic base64 encoded authorization token)

curl -H "Authorization: Basic <base64 encoded
tenant_oauth_app_client_id:tenant_oauth_app_client_secret>"

 -H "Content-Type: application/x-www-form-urlencoded;charset=UTF-8"  --
request POST

https://<idcs-URL.identity>.<TENANT>.com/oauth2/v1/token

 -d
"grant_type=password&scope=urn:opc:idm:__myscopes__&username=<username>&passw
ord=<password>"

echo -n "<PROD_APP_ID_INTERNAL>" | base64 -w 0

curl -H "Authorization: Basic <Basic_Auth_Token>
```

```
 -H "Content-Type: application/x-www-form-urlencoded;charset=UTF-8"  --
request POST

https://<idcs-URL.identity.TENANT.com/oauth2/v1/token

 -d
"grant_type=password&scope=urn:opc:idm:__myscopes__&username=<UserName>
&password=<Password>"
```

After generating the Access Token, invoke the API as shown in the following section.

3.2.4 Invoke the API using the Access Token

To invoke the API using the generated Access Token, do as shown in the following example using cURL Commands in the CLI Tool:

```
curl -iL -H "Authorization: Bearer <access token>" -H "Content-Type:
<content_type>" -d "<request_body>" --cacert <certificate(.pem)> -X
<http_verb> <api_url>

curl -iL -H "Authorization: Bearer <AUTH_TOKEN>"

-H "Content-Type: application/json" -d "{\"type\":\"files\",\"data\":
[{\"fileName\":\"testtoken\",\"mimeType\":\"text/plain\",\"fileSize\":

123}]}" --cacert outcert.pem -X POST https://<OCI-URL>/
<TENANT><APP_ID>/dsa/utils/getObjStoreParUrl
```

3.3 Supported Methods

- **GET**: Retrieve information about the service instance.
- **POST**: Create, scale, backup, start, and stop the service instance.

3.3.1 Media Types

The following media type is supported by the Accounting Foundation Cloud Service REST APIs:

- application/json

3.4 Supported Headers

The REST API supports headers that may be passed in the header section of an HTTP Request or Response.

Table 3-1 Table: Supported Headers

Headers	Description	Example
Content-Type	The media type of the body of the request. Required for POST and PUT requests, and the supported types vary with each endpoint.	Content-Type: application/json
Accept	The media type of the body of the response.	Accept: application/json

3.5 Status Code

When you call the Accounting Foundation Cloud Service REST APIs Resources, the Response Header returns one of the standard HTTP Status Codes.

Table 3-2 Status Code

HTTP Status Code	Description
`200 OK`	The request was successfully completed. A 200 status is returned for a successful GET or POST Method.
`201 Created`	The request has been fulfilled and resulted in a new resource being created. The response includes a Location Header containing the canonical URI for the newly created resource. A 201 status is returned from a synchronous resource creation or an asynchronous resource creation that was completed before the response was returned.
`202 Accepted`	The request has been accepted for processing, but the processing has not been completed. The request may or may not eventually be acted upon, as it may be disallowed at the time the processing takes place. When specifying an Asynchronous (__detached=true) Resource creation (for example, when deploying an application), or update (for example, when redeploying an application), a 202 is returned if the operation is still in progress. If __detached=false, a 202 may be returned if the underlying operation does not complete in a reasonable amount of time.
`400 Bad Request`	The request could not be processed because it contains missing or invalid information (such as a validation error on an input field, a missing required value, and so on).
`401 Unauthorized`	The request is not authorized. The Authentication Credentials included with this request are missing or invalid.

Table 3-2 (Cont.) Status Code

HTTP Status Code	Description
403 Forbidden	The user cannot be authenticated. The user does not have the authorization to perform this request.
404 Not Found	The request includes a resource URI that does not exist.
405 Method Not Allowed	The HTTP verb specified in the request (DELETE, GET, POST, PUT) is not supported for this request URI.
406 Not Acceptable	The resource identified by this request is not capable of generating a representation corresponding to one of the media types in the Accept Header of the request. For example, the client's Accept Header request XML be returned, but the resource can only return JSON.
409 Conflict	The client's ContentType Header is not correct (for example, the client attempts to send the request in XML, but the resource can only accept JSON).
415 Not Acceptable	The client's ContentType Header is not correct (for example, the client attempts to send the request in XML, but the resource can only accept JSON).
500 Internal Server Error	The server encountered an unexpected condition that prevented it from fulfilling the request.
503 Service Unavailable	The server is unable to handle the request due to temporary overloading or maintenance of the server. The REST Web Application is not currently running.

4
Generating PAR URL for File Operations

The PAR URL for File Operations API creates a PAR File that you can use to perform file operations in the Object Store for end-to-end integrations.

4.1 Generating PAR URL for File Upload

You can use this REST API to generate the PAR URL for File Upload. See the following sections for information on how to perform the post operation.

4.1.1 End Point Details

- **Method** – POST
- **URL** - `https://<HOST_NAME:PORT>/<TENANT>/dsa-ext/v1/fileupd/gen-par`
- **Content-Type** - Application/Json

4.1.2 Calling the API to Generate the URL

To call the API, follow these steps:

1. Open a relevant tool.
2. Copy only the Authorization Token that you got from the Authentication Request.
3. Go to Header.
4. Enter KEY as an **Authentication** and Description as a bearer (Authorization Token) `<eyJ4NXQjUzI1NiI6Ikk3cWxndm1Kal...>`.
5. Send a request using the **POST** Method. The request must be in the following format:
   a. **URL**- https://<HOST_NAME:PORT>/<TENANT>/dsa-ext/v1/fileupd/gen-par
   b. **Query Parameter** - No query parameters.

4.1.2.1 Request JSON Parameters

This section provides the list of parameters in the JSON Request.

Table 4-1 Request JSON Parameters

Name	Type	Required	Description
fileName	STRING	Yes	The name of the file to be uploaded. The following are the conditions for to enter in this field: • Must start with an Alphanumeric Character • Allowed characters are alphabets, numbers, and special characters - hyphen(-), dot(.), and underscore(_) • Length of characters must not be greater than 255 characters
fileSize	INTEGER	Yes	The size of the file to be uploaded. The size of the file should be greater than 1 Byte and less than 7 GB.
mimeType	STRING	Yes	The mime type to be uploaded. The following mime types are allowed: • text/csv • text/plain • application/vnd.openxmlformats-officedocument.spreadsheetml.sheet

Request JSON Sample

```
 [{
"fileName": "File.csv",
"fileSize": 7654,
"mimeType": "text/csv"
}, {
"fileName": "File1.csv",
"fileSize": 8765,
"mimeType": "text/csv"
}]
```

4.1.2.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

Table 4-2 POST JSON Response

Name	Type	Description
code	STRING	The Status Code. The following is the list with details: • DS0000 - Success • DS0002- Service Not Found • DS0003 - Invalid JSON • DS0004 - Request Body is empty • DS0005 - File Name cannot be empty • DS0006 - File name [{0}] length cannot be greater than 255 characters • DS0007 - MIME type cannot be empty for {0} • DS0008 - Invalid MIME type specified for {0}. Supported MIME types : {1} • DS0009 - File size not specified for {0} • DS0010 - File size should be greater than 1B and less than 7GB for {0} • DS0012 - File {0} must start with an alphanumeric character and allowed characters in file name are alphanumeric characters, hyphen(-), dot(.), underscore(_)")
fileName	STRING	The name of the file to be uploaded.
uploadURL	STRING	The generated pre-authenticated URL to upload a file.
fileId	INTEGER	The unique File Identifier.

Response JSON Sample

```
{
"code": "DS0000",
"body": {
"success": [{
"filename": "File.csv",
"uploadURL":
        "https://objectstorage.us-phoenix-1.oraclecloud.com/p/
t7iU0y7009NXbFpXozqB5EJKvUv25VfzoK1j8e3yK_GgfHHeW9sCcWMdMI_IVRg4/n/
```

```
oraclegbudevcorp/b/fsgbu_fsafnd_cndevcorp_h3ofin-prd_default/o/
default/ett/wti/335fbbef-e160-4825-8974-3bf957c6dc00",
"fileId": 34034
}, {
"filename": "File1.csv",
"uploadURL":
        "https://objectstorage.us-phoenix-1.oraclecloud.com/p/
e0xuvdJw6zSFF00hO-I1kwgko624ouF5CNevjTpWMSYpK3180CYJ4Xhkpx4P33cp/n/
oraclegbudevcorp/b/fsgbu_fsafnd_cndevcorp_h3ofin-prd_default/o/
default/asg/seo/f4ed02e4-581d-46a7-8ebc-c101261eeaf9",
"fileId": 34033
}],"failed": []
}
}
```


4.2 Generating PAR URL For File Download

You can use this REST API to generate the PAR URL for File Download. See the following sections for information on how to perform the post operation.

4.2.1 End Point Details

- **Method** – POST
- **REST Endpoint** - /dsa-ext/v1/filedwd/gen-par
- **Content-Type** - Application/Json

Table 4-3 End Point Details

Name	Type	Required	Value
`Content-Type`	String	YES	application/json

4.2.1.1 Request JSON Parameters

Table 4-4 Request JSON Parameters

Name	Type	Required	Description
`fileName`	STRING	YES	The name of the file to be downloaded and the supported formats are CSV, XLSX, and TXT. **For Example**: `ah_acct_period_16-Nov-2022.csv`

Request JSON Sample

```
{"ah_acct_period_16-Nov-2022.csv"}
```

4.2.1.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

Table 4-5 Table: POST JSON Response

Name	Type	Description
`fileName`	STRING	The name of the file to be downloaded.
`downloadURL`	STRING	The generated pre-authenticated URL to download a file.

> **Note:**
>
> The downloadURL is displayed only when the pre-authenticated URL exists and status of the file is **Available**.

Table 4-5 (Cont.) Table: POST JSON Response

Name	Type	Description
code	STRING	The Status Code. The following is the list with details: • DS0000 - Success • DS0004 - Request Body is empty • DS0005 - File Name cannot be empty

Response JSON Sample

```
{
    "code": "DS0000",
    "body": {
        "success": [
            {
                "fileName": "ah_acct_period_16-Nov-2022.csv",
                "downloadURL":
      "https://objectstorage.us-phoenix-1.oraclecloud.com/p/
ywJMqilN7geicAnRKLuFaX_UaGhVzBohk9fFe6i8EgWN6y6d__2fyN1K-ssixTeZ/n/
oraclegbudevcorp/b/fsgbu_fsafnd_cndevcorp_qfgnhy-prd_default/o/
default/jfm/cdq/761dac86-e187-495c-92cc-4f1760dfe839"
}
],
"failed": []
    }
}
```

5
Initiating a PMF Process

This section provides information on how to initiate a PMF process.

5.1 End Point Details

- **Method** – POST
- **REST Endpoint** - /pmf-exec/rest-api/v1/PMFService/execute
- **Content-Type** - Application/Json

Table 5-1 End Point Details

Name	Type	Required	Value
`ofs_service_id`	String	Yes	`OFS_FSAFND`
`Content-Type`	String	No	`application/json`

After the pipeline is executed, `responseStatus` indicates "`SUCCESS`" or "`failed`" based on the result of the execution.

5.1.1 Request JSON Parameters

Table 5-2 Request JSON Parameters

Name	Type	Required	Description
`SummaryPayload`	OBJECT	Yes	This is an object with the properties mentioned below
`objectid`	STRING	Yes	Specify an unique string identifier for the execution. Allowed characters are alphabets , numbers, Hyphen(-),and Underscore(_) No other special characters are allowed For example: 'test123'
`processID`	STRING	Yes	Process ID of the Run Pipeline
`objecttype`	STRING	NO	Specify an empty string
`locale`	STRING	Yes	Specify values as 'en-us'
`applicationparams`	OBJECT	Yes	Specify values for the run time execution parameters.

Request JSON Sample

```
{
"SummaryPayload": {
    "objectid": "$objectId",
    "processID": "1652966081449",
    "objecttype": "null",
    "locale": "en-US",
    "applicationparams":{"FIC_MIS_DATE":"2020-06-01"}
  }
}
```

5.1.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

Table 5-3 Table: POST JSON Response

Name	Type	Description
processInstanceID	STRING	Instance Id of the Process being executed
stopExecutionFlag	STRING	Flag indicates if the execution stopped due to internal failure
processStatus	STRING	It indicates if the process execution has started Successfully/Failed
processID	STRING	Process ID of the Run Pipeline
workflowStatus	STRING	Status of the process execution
lastActivity	STRING	It is the next activity ID in the process to be executed
responseJSON	STRING	Response of a particular task, if any
responseStatus	STRING	Rest API standard status

Response JSON Sample

```
{
        "payload": {
        "processInstanceID": "1634536080515_3f80680b-3276-42ac-a41c-
a573d0697c23",
        "stopExecutionFlag": false,
        "processStatus": "SUCCESS",
        "processID": "1634536080515",
        "workflowStatus": "COMPLETED",
        "lastActivity": "Job_1583994521890",
        "responseJSON": "",
        "responseStatus": "SUCCESS"
    }
```

6
Viewing Ingested Data

The APIs allow you to view the data loaded into the system using the Ingestion Connector. You are required to follow the two-step process described in this topic. The first step is to use the Metadata Keys API to fetch the Metadata Keys. The second step is to use the Metadata Keys in the Retrieve Data using the Metadata Keys API.

6.1 Prerequisites

The following are the prerequisites to use the APIs listed in this section:

- Users must be mapped to the Data Access User Group (AFCSDTACCGRP).
- Users must get the Connector ID.
  For more information on how to get the Connector ID, see the below links
  - Getting the Connector Metadata Key from the Data Catalog Viewer
  - Getting the Connector Metadata Key from the Connector Details page
- Users must have all the details required to invoke the APIs.
  For more information, see Executing Pipeline Using an API.

6.2 Get Metadata Keys

Use the Get method to fetch the Metadata Keys. The Get Metadata Keys API enables you to fetch the Business Term Codes for the Target that is used in the Connector. You can use the Business Term Codes in the Retrieve Data Using the Metadata Keys API request to retrieve data.

6.2.1 End Point Details

- **Method** – GET
- **URL** - `https://<HOST_NAME:PORT>/<TENANT>/dsa-ext/v1/data-access/keys/<connector_id>`
- **Content-Type** - application/json

6.3 Calling the API to Generate the URL

To call the API, follow these steps:

1. Open a relevant tool.
2. Copy only the Authorization Token that you got from the Authentication Request.
3. Go to Header.
4. Enter KEY as an **Authentication** and Description as a bearer (Authorization Token) `<eyJ4NXQjUzI1NiI6Ikk3cWxndm1Kal...>`.

5. Send a request using the **GET** Method. The request must be in the following format:
   - **URL**- `https://<HOST_NAME:PORT>/<TENANT>/dsa-ext/v1/data-access/keys/<connector_id>`

6.3.1 Request JSON Parameters

There is no payload to be submitted during API invocation.

6.3.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

Table 6-1 POST JSON Response

Name	Type	Description
name	STRING	Connector name.
code	INTEGER	Connector Metadata Key
attributes	ARRAY	Business Term Attributes · name - The Business Term Name. This is a STRING. · code - The Business Term Code. This is a STRING.

Response JSON Sample

```
{

"code": "DIH0000",

"body": {

"name": "EXAMPLE_CONN_CDS_MT",

"code": "820",

"attributes": [{

"name": "As Of Date",

"code": "BTO1375"

}, {

"name": "Joining Credit Score",

"code": "BTO2010"

}, {

"name": "Original Account Number",
```

```
"code": "BTO2726"

}, {

"name": "Account Or Contract Number",

"code": "BTO53"

}, {

"name": "Address Identifier",

"code": "BTO126"

}, {

"name": "Address Line One",

"code": "BTO127"

}, {

"name": "As Of Date",

"code": "BTO1375"

}, {

"name": "Account Or Contract Numeric Identifier",

"code": "BTO3956"

}, {

"name": "Address Numeric Identifier",

"code": "BTO3959"

}]

}

}
```


6.4 Retrieve Data Using the Metadata Keys

Pass the Business Term Codes received in the response from the Get Metadata Keys API in the Retrieve Data using the Metadata Keys API request to retrieve data. You can specify a filter. However, the filter expression allows only one column.

The API retrieves the JSON response payload data up to 5MB.

6.4.1 End Point Details

- **Method** – POST
- **URL** - `https://<HOST_NAME:PORT>/<TENANT>/dsa-ext/v1/data-access/data/<connector_id>`
- **Content-Type** - application/json

6.4.2 Calling the API to Generate the URL

To call the API, follow these steps:

1. Open a relevant tool.
2. Copy only the Authorization Token that you got from the Authentication Request.
3. Go to Header.
4. Enter KEY as an Authentication and Description as a bearer (Authorization Token) `<eyJ4NXQjUzI1NiI6Ikk3cWxndm1Kal...>`.
5. Send a request using the POST Method. The request must be in the following format:
   - **URL** - `https://<HOST_NAME:PORT>/<TENANT>/dsa-ext/v1/data-access/data/<connector_id>`

6.4.2.1 Request JSON Parameters

Table 6-2 JSON Request

Name	Type	Required	Description
attr_filter	STRING	Yes	Business Term Codes from the Get Metadata Keys API. NOTE: · The Business Term Codes must be enclosed within square brackets ([ ]).
data_filter	OBJECT	Yes	Key is the Business Term Code and the value is the corresponding value for the Business Term Code. For example: "data_filter": {"BTO1375": "01-Apr-2022"} NOTE: · The filter expression allows only one column.

Request JSON Sample

```
{

    "attr_filter": ["BTO1375", "BTO2010", "BTO2726", "BTO53", "BTO126",
"BTO127", "BTO1375", "BTO3956", "BTO3959"],

    "data_filter": {"BTO1375": "01-Apr-2022"}

}
```

6.4.2.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

Table 6-3 POST JSON Response

Name	Type	Description
name	STRING	Logical Entity Name used in the Connector.
data	ARRAY	Key is the Business Term Code and the value is the corresponding value present in the table for the Business Term Code.

Response JSON Sample

```
{

"code": "DIH0000",

"body": [{

"name": "Account",

"data": [{

"BTO53": "5",

"BTO3956": "5",

"BTO1375": "2022-04-01 00:00:00",

"BTO2726": " ",

"BTO2010": " "

}, {

"BTO53": "55",

"BTO3956": "55",
```

```
"BTO1375": "2022-04-01 00:00:00",

"BTO2726": " ",

"BTO2010": " "

}]

}, {

"name": "Address",

"data": [{

"BTO3959": "5",

"BTO1375": "2022-04-01 00:00:00",

"BTO126": "abcd",

"BTO127": " "

}, {

"BTO3959": "55",

"BTO1375": "2022-04-01 00:00:00",

"BTO126": "abcde",

"BTO127": " "

}]

}]

}
```

7
Abort PMF Execution Process

This section provides information on how to abort a PMF Execution process. This API aborts execution for given process Instance ID.

7.1 End Point Details

- **Method** – POST
- **REST Endpoint** - /pmf-exec/rest-api/v1/PMFService/abortprocess
- **Content-Type** - Application/Json

Table 7-1 End Point Details

Name	Type	Required	Value
`processInstanceId`	String	Yes	`"1658311255081_fa6a579a-d2f2-4455-8be3-f8b35022ba7f"`

After the pipeline is executed, status indicates “`true`” or “`false`” based on the result of the execution.

7.1.1 Request JSON Parameters

Table 7-2 Request JSON Parameters

Name	Type	Required	Description
processInstanceId	STRING	Yes	Instance Id of the Process being aborted

Request JSON Sample

```
{
    "processInstanceId":"1661750871424_96680f3d-8fd6-4602-86e7-38bbcded6e60"


}
```

7.1.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

Table 7-3 POST JSON Response

Name	Type	Description
processInstanceID	STRING	Instance Id of the Process being aborted
message	STRING	It indicates the process message description For example: Process is already running
status	STRING	Status of the process execution. · Success: True · Failed: False

Table 7-4 Possible Status Codes

Status Code	Description
200	Process is aborted
200	Process is already completed or failed, so the process cannot be cancelled
200	Process Instance ID is not passed
500	Not valid request
401	Unauthorized request

Response JSON Sample

```
{
            "message": "Process instance id is not passed",
            "status": false
          }
```

8
Resume PMF Execution Process

This section provides information on how to Resume a PMF Execution process. This API resumes the process which is in Failed or Cancelled state.

8.1 End Point Details

- **Method** – POST
- **REST Endpoint** - /pmf-exec/rest-api/v1/PMFService/resumeprocess
- **Content-Type** - Application/Json

Table 8-1 End Point Details

Name	Type	Required	Value
processInstanceId	String	Yes	"1658311255081_fa6a579a-d2f2-4455-8be3-f8b35022ba7f"

After the pipeline is executed, status indicates “true” or “false” based on the result of the execution.

8.1.1 Request JSON Parameters

Table 8-2 Request JSON Parameters

Name	Type	Required	Description
processInstanceId	STRING	Yes	Instance Id of the Process being resumed

Request JSON Sample

```
{
    "processInstanceId":"1661750871424_96680f3d-8fd6-4602-86e7-38bbcded6e60"


}
```

8.1.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

Table 8-3 POST JSON Response

Name	Type	Description
processInstanceID	STRING	Instance Id of the Process being resumed
message	STRING	It indicates the process message description For example: Process is already running
status	STRING	Status of the process execution · Success: True · Failed: False

Table 8-4 Possible Status Codes

Status Code	Description
200	Process is resume
200	Process is already running
200	Process is already completed, so the process cannot be resumed
400 bad request	Process Instance ID is not passed
500	PMF Execution service issue
401	Unauthorized request

Response JSON Sample

```
{
        "message": "Process is Resume",
        "status": true
    }
```

9
Rerun PMF Execution Process

This section provides information on how to Rerun a PMF Execution process. This API enables to rerun the triggered PMF execution for a given process Instance ID.

9.1 End Point Details

- **Method** – POST
- **REST Endpoint** - /pmf-exec/rest-api/v1/PMFService/rerunprocess
- **Content-Type** - Application/Json

Table 9-1 End Point Details

Name	Type	Required	Value
`processInstanceId`	String	Yes	`"1658311255081_fa6a579a-d2f2-4455-8be3-f8b35022ba7f"`

After the pipeline is executed, status indicates “`true`” or “`false`” based on the result of the execution.

9.1.1 Request JSON Parameters

Table 9-2 Request JSON Parameters

Name	Type	Required	Description
`processInstanceId`	STRING	Yes	Instance Id of the Process being Rerun

Request JSON Sample

```
{
    "processInstanceId":"1661750871424_96680f3d-8fd6-4602-86e7-38bbcded6e60"


}
```

9.1.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

Table 9-3 POST JSON Response

Name	Type	Description
processInstanceID	STRING	Instance Id of the Process being Rerun
message	STRING	It indicates the process message description For example: Process has been run
status	STRING	Status of the process execution. • Success: True • Failed: False

Table 9-4 Possible Status Codes

Status Code	Description
200 ok	Successful response (a new process instance id will be shared for rerun process)
400 bad request	Process ID or Process Instance ID is not passed
500	PMF execution service issue
401	Unauthorized request
200	Wrong Process Instance ID

Response JSON Sample

```
 {
            "PROCESS_INSTANCE_ID": "1661750871424_7ab3f4a7-4eef-409e-
aa9c-6345bb9e4a4e",
            "message": "Process has been Rerun",
            "status": true
         }
```

10
Fetching the PMF Execution status

This section provides information on how to fetch a PMF Execution status. This API helps to fetch various execution statuses for a given PMF Process execution for a given process Instance ID. The statuses can contain one or more values such as COMPLETED/RUNNING/CANCELLED/FAILED.

10.1 End Point Details

- **Method** – POST
- **REST Endpoint** - /pmf-exec/rest-api/v1/PMFService/execute/status
- **Content-Type** - Application/Json

Table 10-1 End Point Details

Name	Type	Required	Value
processInstanceId	String	Yes	"1658311255081_fa6a579a-d2f2-4455-8be3-f8b35022ba7f"

After the pipeline is executed, status indicates “RUNNING” , “FAILED”,"CANCELLED” or "COMPLETED",based on the result of the execution.

10.1.1 Request JSON Parameters

Table 10-2 Request JSON Parameters

Name	Type	Required	Description
processInstanceId	STRING	Yes	Process Instance ID of the PMF Execution

Request JSON Sample

```
{

  "processInstanceIdList": [

    "1658311255081_fa6a579a-d2f2-4455-8be3-f8b35022ba7f",

    "1658313898588_cf0ed63a-8176-495b-ba15-633a33cf9d0c"

  ]
```

```
}
```

10.1.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

Table 10-3 POST JSON Response

Name	Type	Description
processInstanceID	STRING	Process Instance ID of the PMF Execution
status	STRING	Status of the PMF Execution

Response JSON Sample

```
 {

  "processExecutionStatus": [

    {

      "process_instance_id": "1658311255081_fa6a579a-d2f2-4455-8be3-
f8b35022ba7f",

      "status": "RUNNING"

    },

    {

      "process_instance_id": "1658313898588_cf0ed63a-8176-495b-
ba15-633a33cf9d0c",

      "status": "FAILED"

    }

  ]

}
```

11
Searching a File

This section provides information on how to perform an advance search of a file.

11.1 End Point Details

- Method – POST
- REST Endpoint - /dsa-ext/v1/fileopt/search-file
- Content-Type - Application/Json

The following table lists the Request Headers

Table 11-1 Request Headers-Searching a File

Name	Type	Required	Value
Content-Type	String	NO	`application/json`

11.1.1 Request JSON Parameters

The following table lists the Request Body.

Table 11-2 Request JSON Parameters - Searching a File

Name	Type	Required	Description
`OrdeByCode`	INTEGER	No	The files are sorted based on the inputs provided: • 1 - Sorts the file name from A to Z • 2 - Sorts the file name from Z to A • 3 - Displays the Recent First files • 4 - Displays the Recent Last files For Example: 3
`offset`	INTEGER	No	Specify a value that will skip the N first rows in a result set before starting to return any rows For Example: 1
`fetchSize`	INTEGER	No	Specify the number of files to be fetched in the result set. For Example: 100

Table 11-2 (Cont.) Request JSON Parameters - Searching a File

Name	Type	Required	Description
nameStr	STRING	No	Specify the name of the file For Example: CASA Statement
executionId	STRING	No	Specify the execution ID of the file
processId	STRING	No	Specify the process ID or Process Instance ID of the file
startDate	STRING	No	Specify the start date of the file in the yyyy-MM-dd HH:mm:ss format For Example: 2022-01-01 11:20:50
endDate	STRING	No	Specify the end date of the file in the yyyy-MM-dd HH:mm:ss format For Example: 2022-09-13 06:39:42
minFileSize	INTEGER	No	Specify the minimum size of the file in bytes For Example:200
maxFileSize	INTEGER	No	Specify the maximum size of the file in bytes For Example: 8053063680

Request JSON Sample

```
{"orderByCode": 3,"offset": 0,"fetchSize": 100,"nameStr":
"test","executionId": "141","processId": "1661867637887","startDate":
"2022-01-01 00:00:00","endDate": "2022-09-13 06:39:42","minFileSize":
0,"maxFileSize": 8053063680}
```

11.1.2 Response JSON Parameters

This section provides the list of parameters in the JSON Response.

The following table provides details for the query parameters returned in the response body.

Table 11-3 POST JSON Response

Name	Type	Description
Id	STRING	The file Identifier
name	STRING	The file name

Table 11-3 (Cont.) POST JSON Response

Name	Type	Description
type	STRING	The file type. The response is either ' text/plain' fox TXT files or 'text/csv' for CSV files
length	STRING	The file size in bytes
status	STRING	The status of the file. The response can be any of the following: INITIALIZED, UPLOADING, or AVAILABLE
createdDate	STRING	The creation date of the file
createdBy	STRING	The user name who has created the file
updatedDate	STRING	The files that were updated on a specific date For Example: 2022-08-30 17:37:16
procId	STRING	The Process ID of the file
executionId	STRING	The Execution ID of the file
procInstanceId	STRING	The Process Instance ID of the file
updatedBy	STRING	The user name who has updated the file

Response JSON Sample


```
{

 "code": "DIH0000",

 "body": [{

 "id": "2",

 "name": "extract_address_master.txt",

 "type": "text/plain",

 "length": "233",

 "status": "AVAILABLE",

 "createdDate": "2022-08-30 17:36:57",

 "createdBy": "ADMIN",

 "updatedDate": "2022-08-30 17:37:16",

 "procId": "1661867637887",

 "executionId": "141",
```

```
 "procInstanceId": "1661867637887_74c5b9c3-9814-46e5-
ba7e-1f48ce443bdb",

 "updatedBy": "filestore"

 }

 ]

 }
```

Glossary

Index