Function Import in SAP OData Service

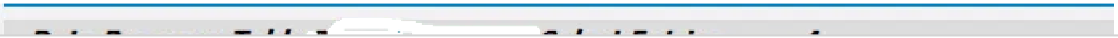
Hello everyone, in this tutorial we will learn how to implement Function Import in SAP OData Service.

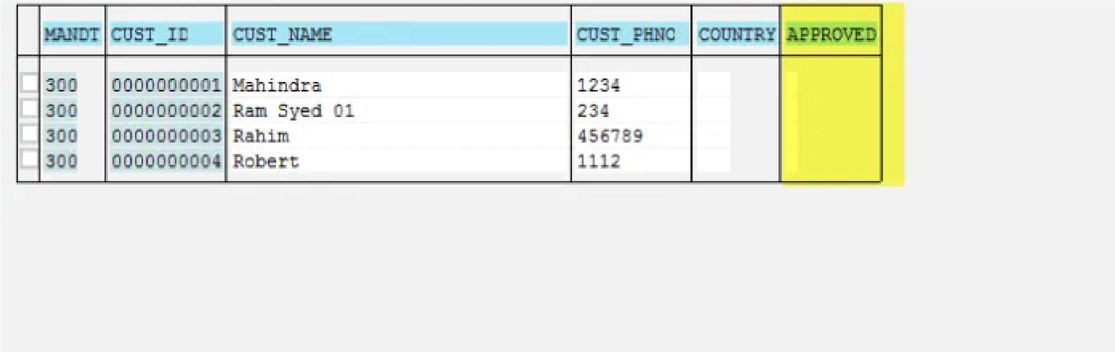
What is Function Import?

Function Imports are the actions like the release of a sales order/purchase order executed in the back-end system. They can be used whenever the given requirement does not fit into the CRUD-Q operations.

They are defined at the service level and you can have N number of function imports functions.

Business Scenario





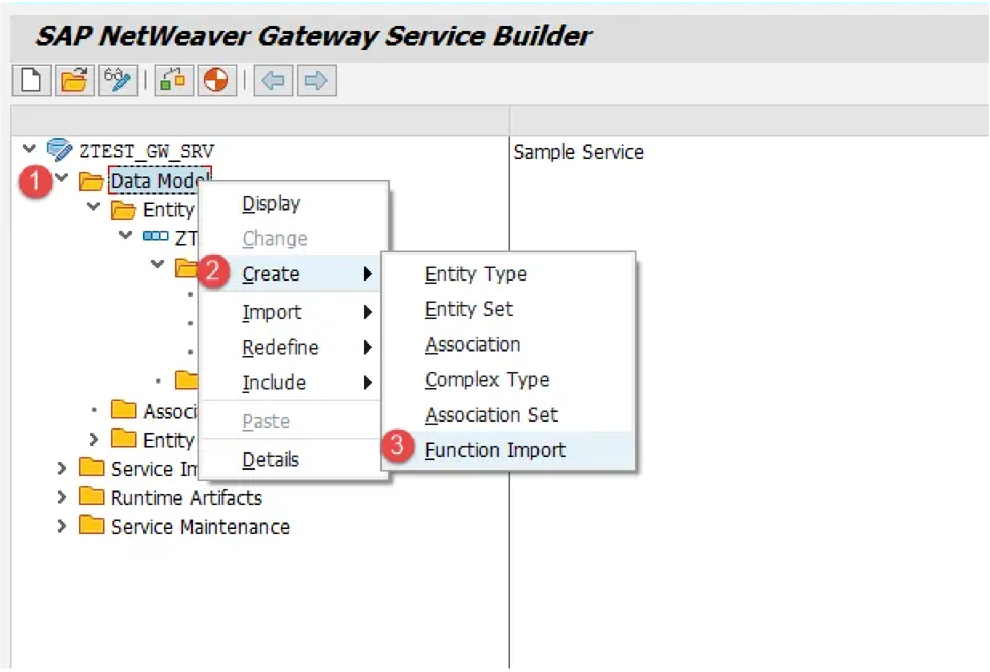
	MANDT	CUST_ID	CUST_NAME	CUST_PHNO	COUNTRY	APPROVED
□	300	0000000001	Mahindra	1234		
□	300	0000000002	Ram Syed 01	234		
□	300	0000000003	Rahim	456789		
□	300	0000000004	Robert	1112		

We have a table in which it contains customer information, two or more customers information have to be approved at once.

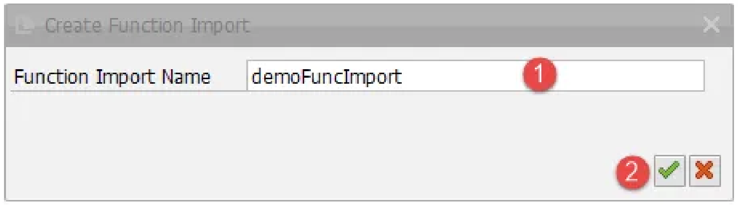
In this scenario to approve each customer information we will use Function Import.

Create Function Import in SAP OData Service

1. Go to Gateway Service Builder **SEGW** and choose the project in which you want to create Function Import.



3. Specify the name of the function import and click on green tick mark button.



Function Import Name: demoFuncImport

4. In the next screen you will find different options for the Function Imports lets have a look at each of them.

Return Type Kind:

If you want the function import to return data you can set this field. We have 3 possible options available

- **Complex Type** – if the return data is of complex entity type, select this option
- **Entity Type** – if the return data is of entity type, select this option and
- **No return** – if you do not want to return anything, select this option.

Return Type:

- If you choose Complex Type in the above step, specify the name of the complex entity type in this field.
- If you choose Entity Type option in the above step, specify the name of entity type.
- If you choose No return, this field will be empty.

Return Cardinality:

‘1’ or ‘n’ based on the Return Type Kind defined above. if ‘n’ is selected provide the entity set name

HTTP Method Type:

Choose the HTTP method type to invoke the Function Import.

- **GET** , when you want to get the data.
- **POST**, when you want to post the data to back-end.

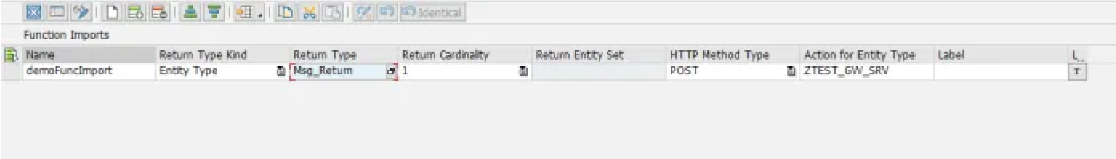
Action for Entity Type:

This field is used to identity to which entity type in the service, the function import is applied as an ACTION. Specify the entity

Label:

Displays object label.

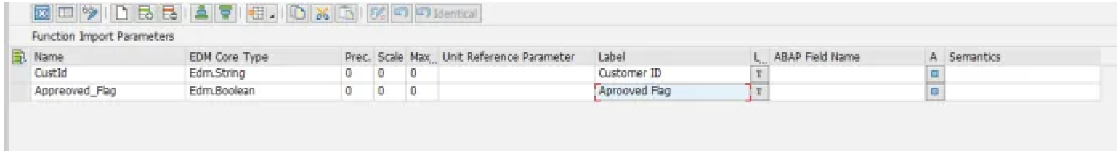
In our business requirement we have set the following options



Function Imports

Name	Return Type Kind	Return Type	Return Cardinality	Return Entity Set	HTTP Method Type	Action for Entity Type	Label	L..
demoFuncImport	Entity Type	Msg_Return	1		POST	ZTEST_GW_SRV		T

5. In this step we need to create Import Parameters to the Function Import we have created in the above step. In our business case, we need to two importing parameters Customer Id and Approval Flag. Create them and **Save** the parameters.



Function Import Parameters

Name	EDM Core Type	Prec.	Scale	Max..	Unit Reference Parameter	Label	L..	ABAP Field Name	A	Semantics
CustId	Edm.String	0	0	0		Customer ID	T			
Appreoved_Flag	Edm.Boolean	0	0	0		Aprooved Flag	T			

6. Till now we have defined the function import and our next step is to implement it before going further we want to make sure that every thing is OK till now.

Save the project and click on Generate to generate run-time objects. We should see the function import in the meta data of

Go to Netweaver Gateway Client **/IWFND/GW_CLIENT** and specify the service name with meta data URI Option and Execute.

Implement Function Import

7. Now we need to implement the function import and this is done in the DPC_EXT class and method is **/IWBEP/IF_MGW_APPL_SRV_RUNTIME~EXECUTE_ACTION.** Go the DPC_EXT class and redefine the method. Put the below code in the method.

```
 DATA: ls_parameter TYPE /iwbep/s_mgw_name_value_pair,
       lv_custid    TYPE kunnr,
       lv_flag      TYPE char1,
       lt_custinfo  TYPE TABLE OF ztest_gw_srv,
       ls_custinfo  TYPE ztest_gw_srv,
       ls_entity    TYPE zcl_ztest_gw_srv_mpc=>ts_msg_return.



     IF iv_action_name = 'demoFuncImport'. " Check what action is being requested
       IF it_parameter IS NOT INITIAL.

 * Read Function import parameter value
         READ TABLE it_parameter INTO ls_parameter WITH KEY name = 'CustId'.
         IF sy-subrc = 0.
           lv_custid = ls_parameter-value.
         ENDIF.

         READ TABLE it_parameter INTO ls_parameter WITH KEY name = 'Approved_Flag'.
         IF sy-subrc = 0.
           lv_flag = ls_parameter-value.
         ENDIF.

         IF  lv_custid IS NOT INITIAL.
           UPDATE ztest_gw_srv SET approved = lv_flag WHERE cust_id = lv_custid.
           IF sy-subrc = 0.
             ls_entity-type   = 'S'.
             ls_entity-message = 'Customer info successfully approved'.
           ELSE.
             ls_entity-type   = 'E'.
             ls_entity-message = 'Error'.
           ENDIF.


 * Call method copy_data_to_ref and export entity set data
           copy_data_to_ref( EXPORTING is_data = ls_entity
                   CHANGING cr_data = er_data ).
         ENDIF.
       ENDIF.
     ENDIF.
```

The above code will update the status of the APPROVED field which exists in the table. If the update is successful it will return the success message or will return the error message.

8. So we are ready with Function Import implementation, it’s time to test the service.

9. Go to Netweaver Gateway Client **/IWFND/GW_CLIENT** and specify the service name with Function Import name and its importing parameters. The Url will look like below

/sap/opu/odata/sap/ZTEST_GW_SRV_SRV/demoFuncImport?Approved_Flag=’X’&CustId=’0000000001'

Change the HTTP method to POST and hit execute button. You should see the success message in output like below.

Go back to table and check whether the approval flag has been set to ‘X’.

Please stay tuned to us for more SAP OData Tutorials. Please feel free to comment and let us know your feedback. Thank you. 🙂

Continue Learning

- What are OData and REST
- Know about the structure of SAP OData Service
- How to activate SAP Netweaver Gateway
- Modeling SAP OData Service
- Implement SAP OData Service
- Register SAP OData Service
- Implement GET_ENTITY method of SAP OData Service
- Association and Navigation in SAP OData Service
- How to perform POST call in SAP OData Service
- List of SAP OData Service Query Options
- SAP OData query – $filter
- SAP OData query – $orderby
- SAP OData query – $top and $skip
- SAP OData query – $inlinecount
- SAP OData query – $skiptoken
- SAP OData query – $expand
- How to implement Deep entity (or) Deep Insert in SAP OData Service
- How to implement ETag in SAP OData Service
- How to consume external OData services or HANA XSOData service
- SAP OData frequently asked interview questions

X