Contents

Chapter 1

ODATA Services

1.1 Goal

1. Understand and Create an ODATA service that interacts with the database doing the following actions:
    1. C - Create
    2. R - Read
    3. U - Update
    4. D - Delete

Chapter 2

Creation of an ODATA service (in the Backend) using the transaction code SEGW

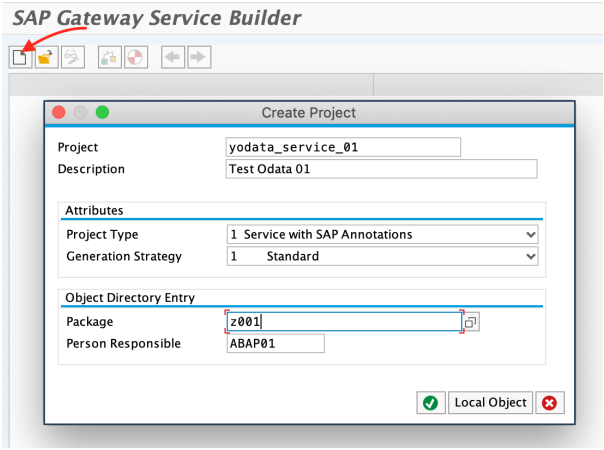
2.1 Prerequisite

1. Table Source (Create your own table, name it YODATA_TAB_##)
2. (Optional) Create Maint View (Use FG YFG_ODATA_##)
3. (Optional) Populate your table with data

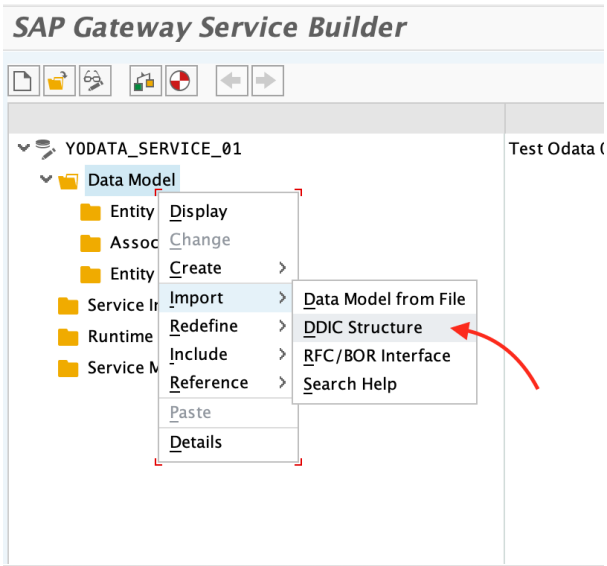
2.2 Create an ODATA service

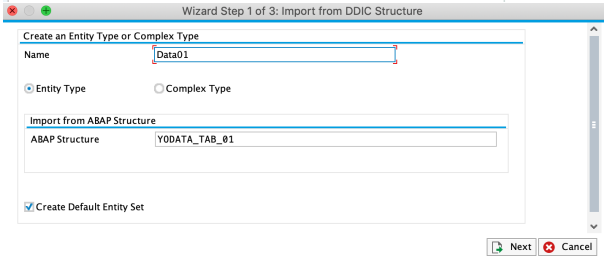
1. Start SEGW

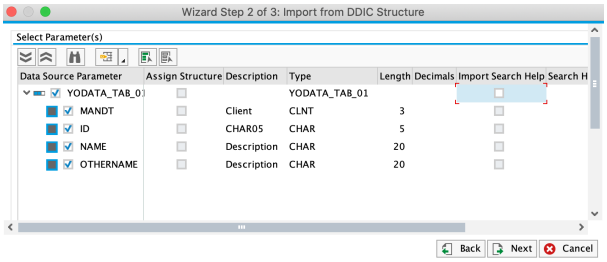
2. Create Project
    1. YODATA_SERVICE_01

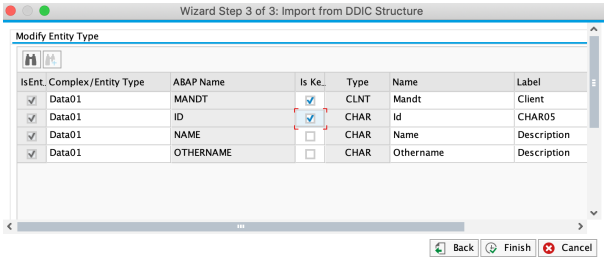


3. Create Data Model (Import from DDIC or Manually Add fields)

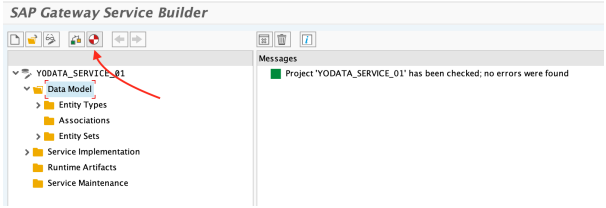


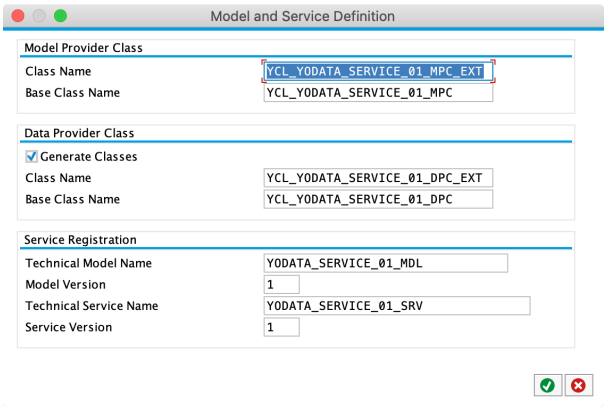






4. Generate Runtime Objects of the Service

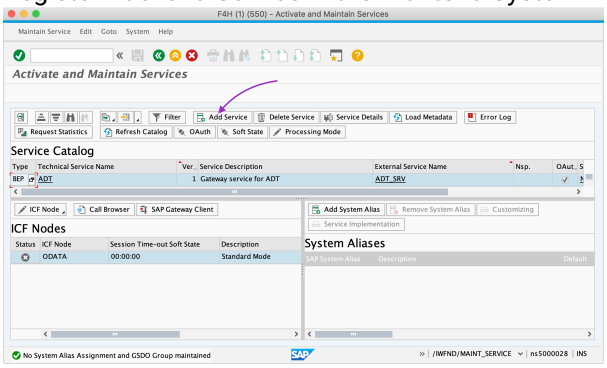



Model and Service Definition
Model Provider Class
Class Name
YCL_YODATA_SERVICE_01_MPC_EXT
Base Class Name
YCL_YODATA_SERVICE_01_MPC
Data Provider Class
Generate Classes
Class Name
YCL_YODATA_SERVICE_01_DPC_EXT
Base Class Name
YCL_YODATA_SERVICE_01_DPC
Service Registration
Technical Model Name
YODATA_SERVICE_01_MDL
Model Version
1
Technical Service Name
YODATA_SERVICE_01_SRV
Service Version
1

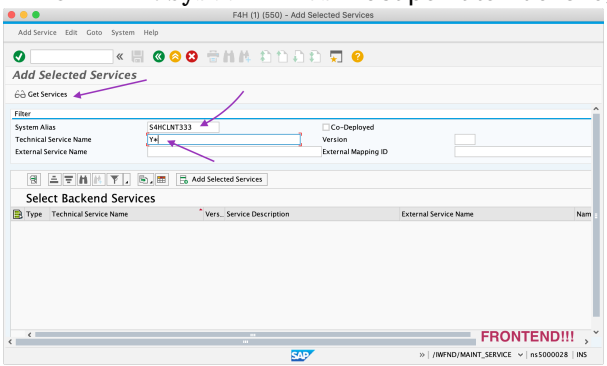
Chapter 3

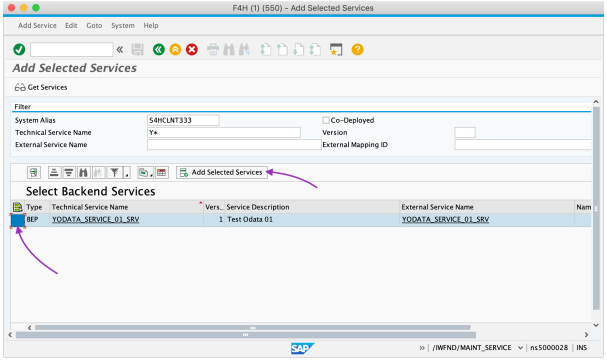
Registration of a service to a system (usually Frontend System)

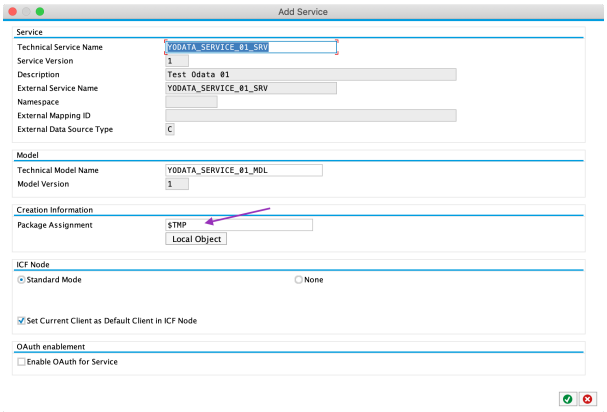
1. Login to the Front End Server (For embedded systems, the frontend and backend reside in the same system installation)
2. Register Backend Service in the Frontend System



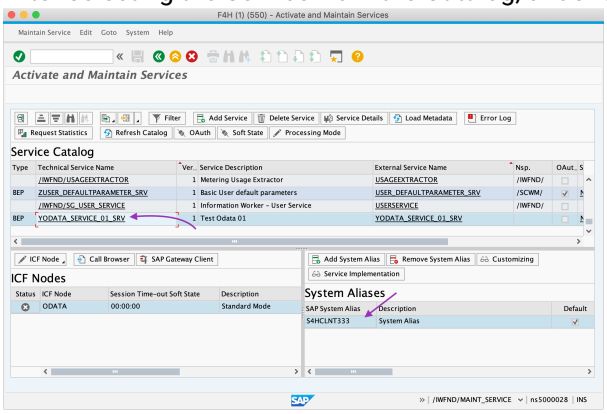
IMPORTANT! `System Alias` must point to Backend, where the data/service is located







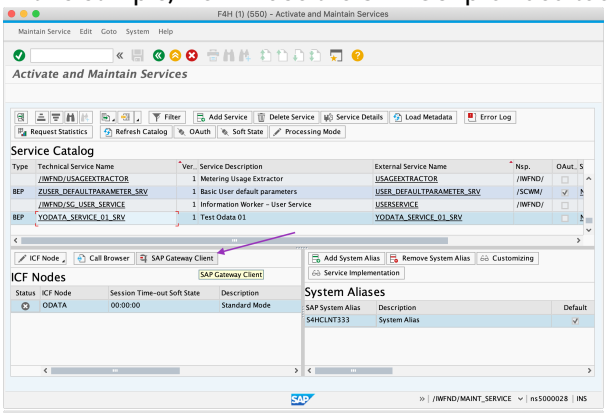
After selecting the Service from the Catalog, under system aliases, the correct backend system should be displayed

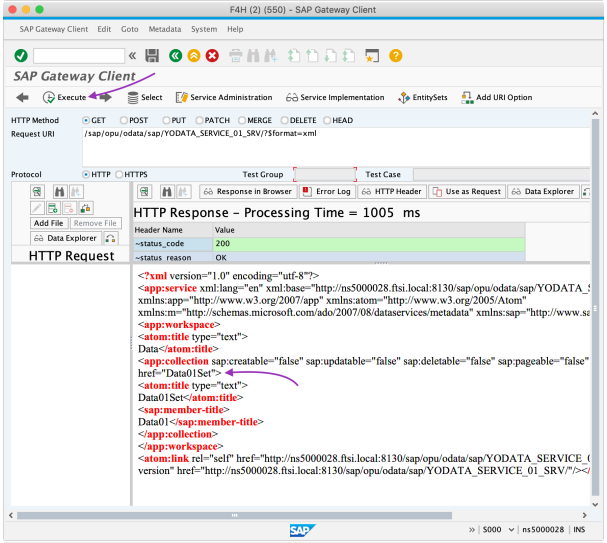


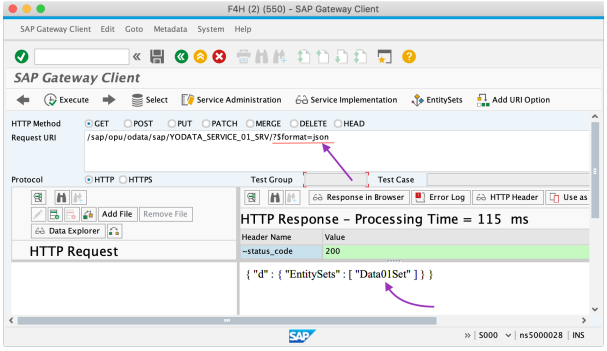
Chapter 4

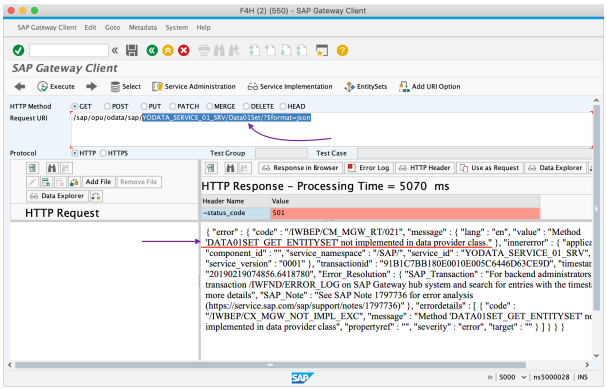
Testing connection

1. We test by using the built in SAP Gateway Client, or by using the browser, or other plugin like Postman
   In this sample, we will use the SAP GUI provided tool.







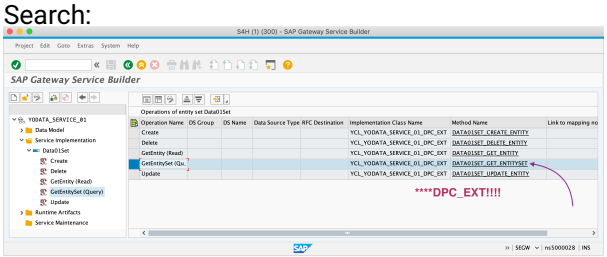

F4H (2) (550) - SAP Gateway Client
SAP Gateway Client Edit Goto Metadata System Help
SAP Gateway Client
Execute Select Service Administration Service Implementation EntitySets Add URI Option
HTTP Method GET POST PUT PATCH MERGE DELETE HEAD
Request URI /sap/opu/odata/sap/YODATA_SERVICE_01_SRV/Data01Set/?$format=json
Protocol HTTP HTTPS
Test Group
Test Case
Add File Remove File
Data Explorer
HTTP Request
Response in Browser Error Log HTTP Header Use as Request Data Explorer
HTTP Response – Processing Time = 5070 ms
Header Name Value
~status_code 501
{ "error" : { "code" : "/IWBEP/CM_MGW_RT/021", "message" : { "lang" : "en", "value" : "Method 'DATA01SET_GET_ENTITYSET' not implemented in data provider class." }, "innererror" : { "applica "component_id" : "", "service_namespace" : "/SAP/", "service_id" : "YODATA_SERVICE_01_SRV", "service_version" : "0001" }, "transactionid" : "91B1C7BB180E0010E005C6446D63CE9D", "timestar "20190219074856.6418780", "Error_Resolution" : { "SAP_Transaction" : "For backend administrators transaction /IWFND/ERROR_LOG on SAP Gateway hub system and search for entries with the timest more details", "SAP_Note" : "See SAP Note 1797736 for error analysis (https://service.sap.com/sap/support/notes/1797736)" }, "errordetails" : [{ "code" : "/IWBEP/CX_MGW_NOT_IMPL_EXC", "message" : "Method 'DATA01SET_GET_ENTITYSET' n implemented in data provider class", "propertyref" : "", "severity" : "error", "target" : "" }] } }
S000 ns5000028 INS

Chapter 5

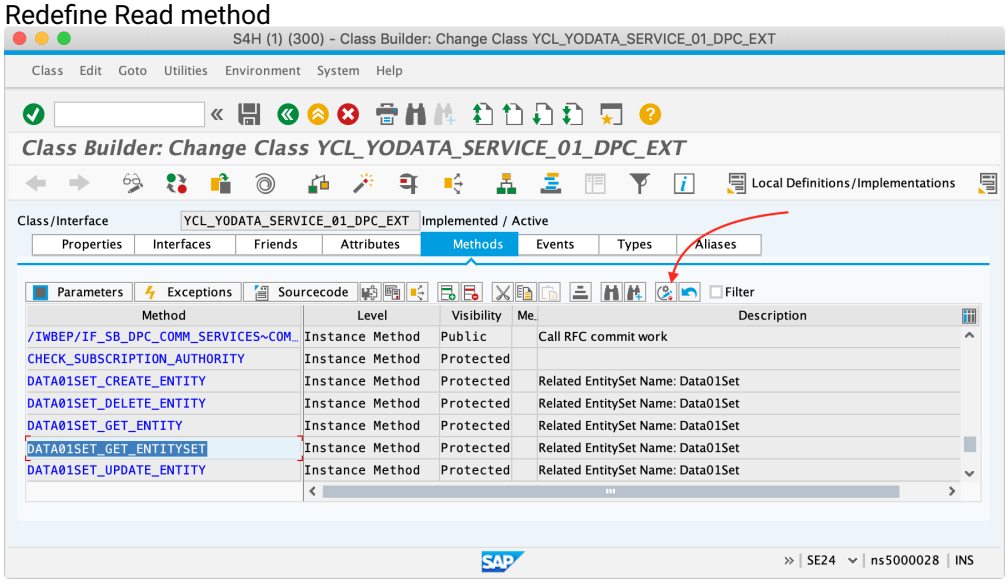
Completing the Implementations

5.1 Class artifact of implementations

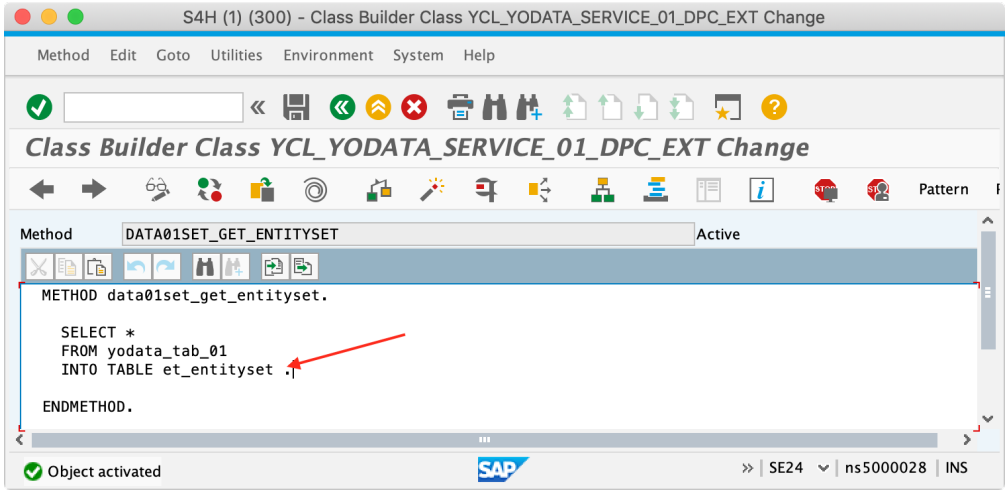
Search:



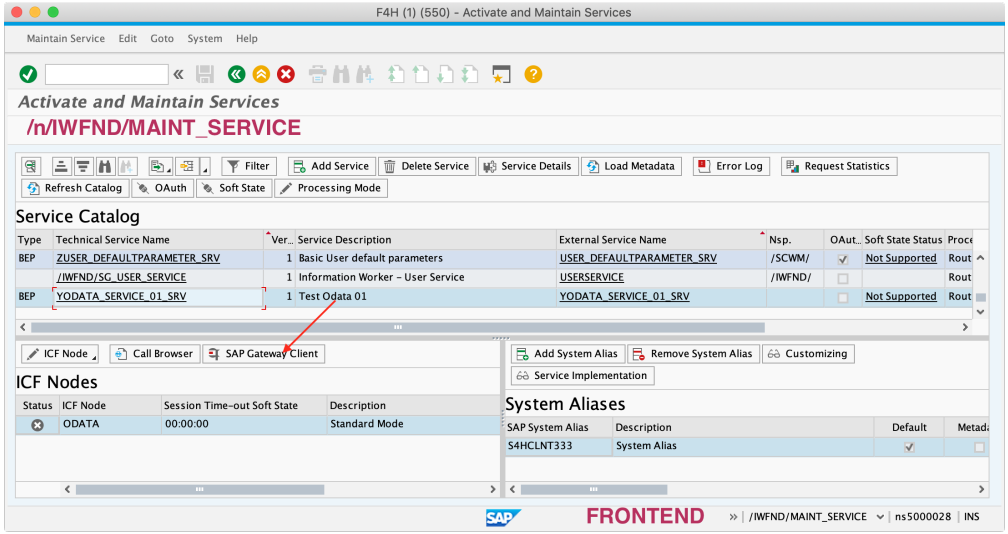
1. Read Data (Multi)
    1. Open the Correct Class (Extension of Data Provider Class)
    2. Redefine Read method

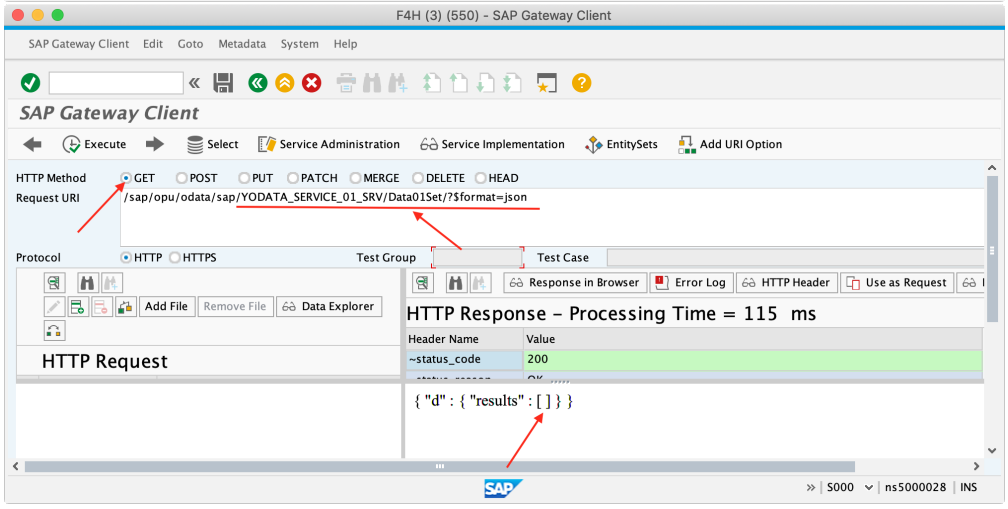


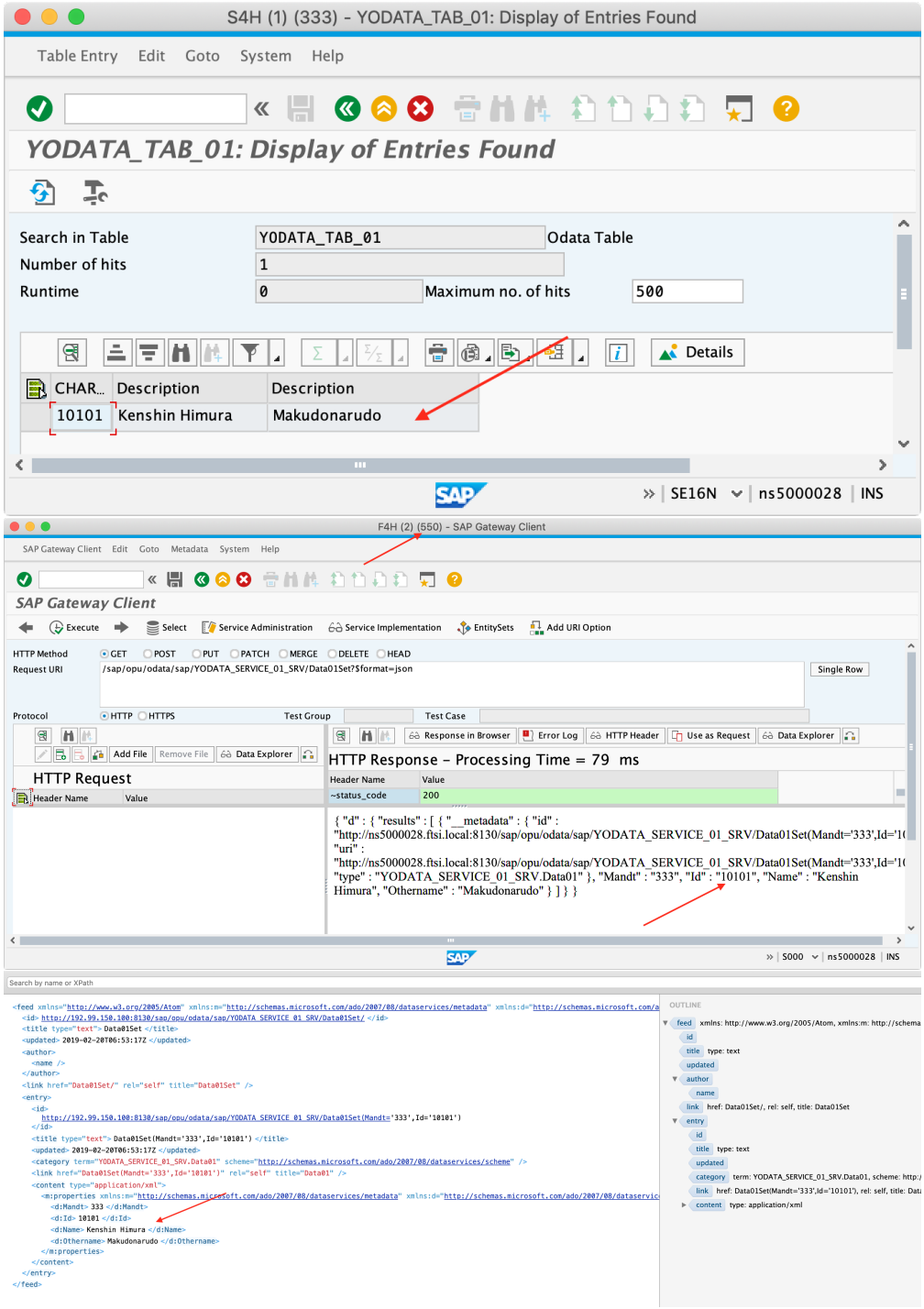
3. Insert Codes for desired behaviour


S4H (1) (300) - Class Builder Class YCL_YODATA_SERVICE_01_DPC_EXT Change
Method Edit Goto Utilities Environment System Help
Class Builder Class YCL_YODATA_SERVICE_01_DPC_EXT Change
Pattern
Method
DATA01SET_GET_ENTITYSET
Active
METHOD data01set_get_entityset.
SELECT *
FROM yodata_tab_01
INTO TABLE et_entityset .
ENDMETHOD.
Object activated
SE24
ns5000028
INS

4. Test


F4H (1) (550) - Activate and Maintain Services
Maintain Service Edit Goto System Help
Activate and Maintain Services
/n/IWFND/MAINT_SERVICE
Filter
Add Service
Delete Service
Service Details
Load Metadata
Error Log
Request Statistics
Refresh Catalog
OAuth
Soft State
Processing Mode
Service Catalog
Type
Technical Service Name
Ver..
Service Description
External Service Name
Nsp.
OAut..
Soft State Status
BEP
ZUSER_DEFAULTPARAMETER_SRV
1
Basic User default parameters
USER_DEFAULTPARAMETER_SRV
/SCWM/
Not Supported
/IWFND/SG_USER_SERVICE
1
Information Worker - User Service
USERSERVICE
/IWFND/
BEP
YODATA_SERVICE_01_SRV
1
Test Odata 01
YODATA_SERVICE_01_SRV
Not Supported
ICF Node
Call Browser
SAP Gateway Client
ICF Nodes
Status
ICF Node
Session Time-out Soft State
Description
ODATA
00:00:00
Standard Mode
Add System Alias
Remove System Alias
Customizing
Service Implementation
System Aliases
SAP System Alias
Description
Default
S4HCLNT333
System Alias
FRONTEND
/IWFND/MAINT_SERVICE
ns5000028
INS


F4H (3) (550) - SAP Gateway Client
SAP Gateway Client Edit Goto Metadata System Help
SAP Gateway Client
Execute
Select
Service Administration
Service Implementation
EntitySets
Add URI Option
HTTP Method
GET
POST
PUT
PATCH
MERGE
DELETE
HEAD
Request URI
/sap/opu/odata/sap/YODATA_SERVICE_01_SRV/Data01Set/?$format=json
Protocol
HTTP
HTTPS
Test Group
Test Case
Add File
Remove File
Data Explorer
Response in Browser
Error Log
HTTP Header
Use as Request
HTTP Request
HTTP Response - Processing Time = 115 ms
Header Name
Value
~status_code
200
{ "d" : { "results" : [ ] } }
S000
ns5000028
INS



2. Read Data (Single)
    1. Open the Correct Class (Extension of Data Provider Class)
    2. Redefine Read method

    3. Insert Codes for desired behaviour

```
METHOD data01set_get_entity.

  DATA: BEGIN OF ls_keys,
          mandt TYPE ycl_yodata_service_01_mpc=>ts_data01-mandt,
          id    TYPE ycl_yodata_service_01_mpc=>ts_data01-id,
        END OF ls_keys.

  READ TABLE it_key_tab ASSIGNING FIELD-SYMBOL(<mandt>)
```

```
    WITH KEY name = 'Mandt'.
    IF sy-subrc IS INITIAL.
      ls_keys-mandt = <mandt>-value.
    ENDIF.

    READ TABLE it_key_tab ASSIGNING FIELD-SYMBOL(<id>)
    WITH KEY name = 'Id'.
    IF sy-subrc IS INITIAL.
      ls_keys-id = <id>-value.
    ENDIF.

    SELECT SINGLE *
    FROM yodata_tab_01
    CLIENT SPECIFIED
    WHERE mandt = @ls_keys-mandt
      AND id = @ls_keys-id
    INTO @er_entity.


  ENDMETHOD.
```

4. Test

1. Create Data
    1. Open the Correct Class (Extension of Data Provider Class)
    2. Redefine Create method
    3. Insert Codes for desired behaviour

```
  METHOD data01set_create_entity.

    DATA: ls_req_payload TYPE ycl_yodata_service_01_mpc=>ts_data01.
    io_data_provider->read_entry_data(
      IMPORTING
        es_data = ls_req_payload ).

    INSERT yodata_tab_01 FROM ls_req_payload.
    COMMIT WORK AND WAIT.

    er_entity = ls_req_payload.
*   Do not forget to add error handling
  ENDMETHOD.
```

4. Test

1. Delete Data
    1. Open the Correct Class (Extension of Data Provider Class)
    2. Redefine Delete method
    3. Insert Codes for desired behaviour

```
  METHOD data01set_delete_entity.

    DATA: BEGIN OF ls_keys,
            mandt TYPE ycl_yodata_service_01_mpc=>ts_data01-mandt,
            id    TYPE ycl_yodata_service_01_mpc=>ts_data01-id,
          END OF ls_keys.

    READ TABLE it_key_tab ASSIGNING FIELD-SYMBOL(<mandt>)
    WITH KEY name = 'Mandt'.
    IF sy-subrc IS INITIAL.
```

```
      ls_keys-mandt = <mandt>-value.
    ENDIF.

    READ TABLE it_key_tab ASSIGNING FIELD-SYMBOL(<id>)
    WITH KEY name = 'Id'.
    IF sy-subrc IS INITIAL.
      ls_keys-id = <id>-value.
    ENDIF.

    SELECT SINGLE *
    FROM yodata_tab_01
    CLIENT SPECIFIED
    WHERE mandt = @ls_keys-mandt
      AND id = @ls_keys-id
    INTO @DATA(ls).
    IF sy-subrc IS INITIAL.
      DELETE yodata_tab_01 FROM ls.
    ENDIF.

  ENDMETHOD.
```

4. Test

1. Update Data
    1. Open the Correct Class (Extension of Data Provider Class)
    2. Redefine Update method
    3. Insert Codes for desired behaviour
       Do it yourself.
    4. Test

Chapter 6

Advanced Topics

6.1 Creation of an ODATA service using a CDS

6.2 Querying Techniques

6.2.1 Use of `$select`, `$top`, `$filter`, `$paginate`, etc

6.3 Testing outside of Gateway SAP Client

1. Browser
1. Postman
1. Curl

6.4 Error Handling

6.5 Authentication

6.6 Encryption

6.7 Browser Debugging

6.8 Create from BAPI