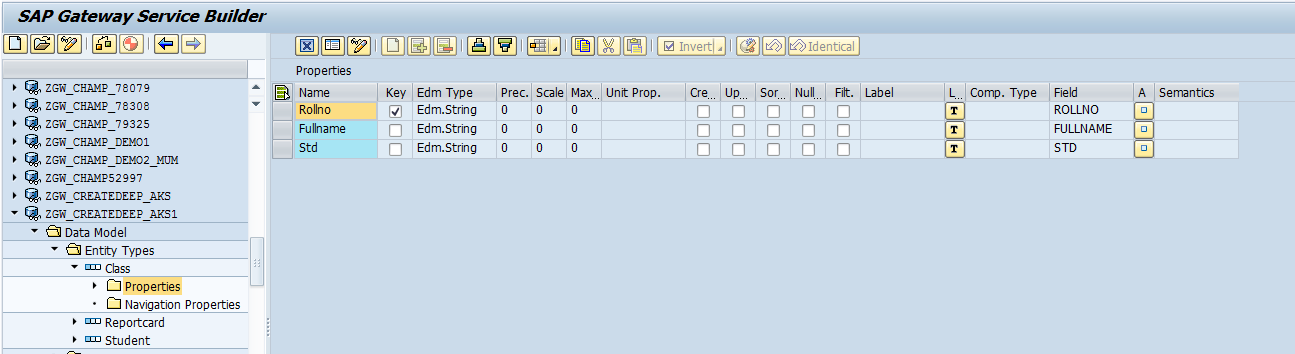
**CREATE DEEP ENTITY**

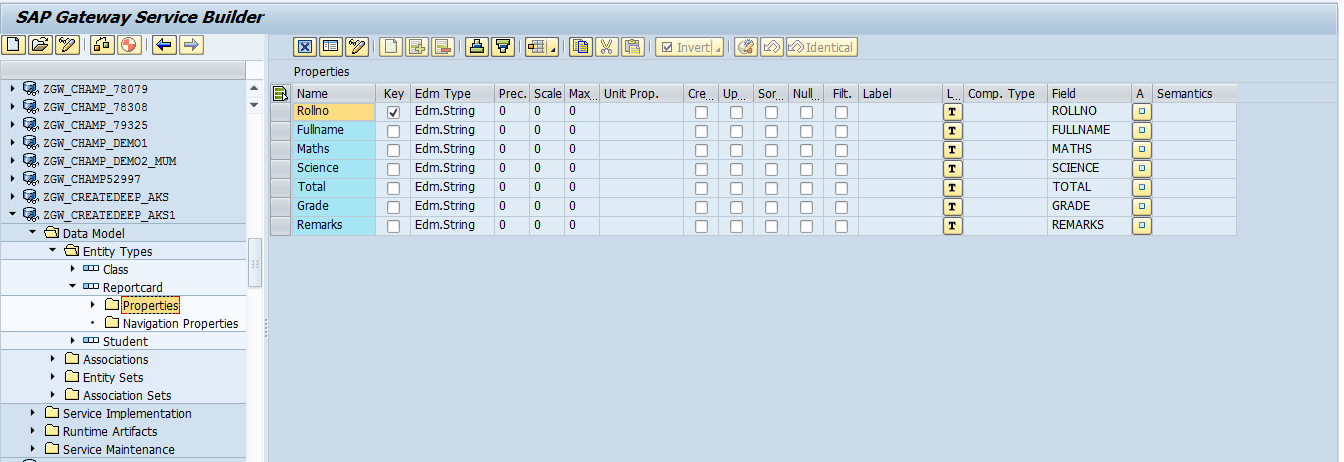
Step 1 :

Create three Entities:

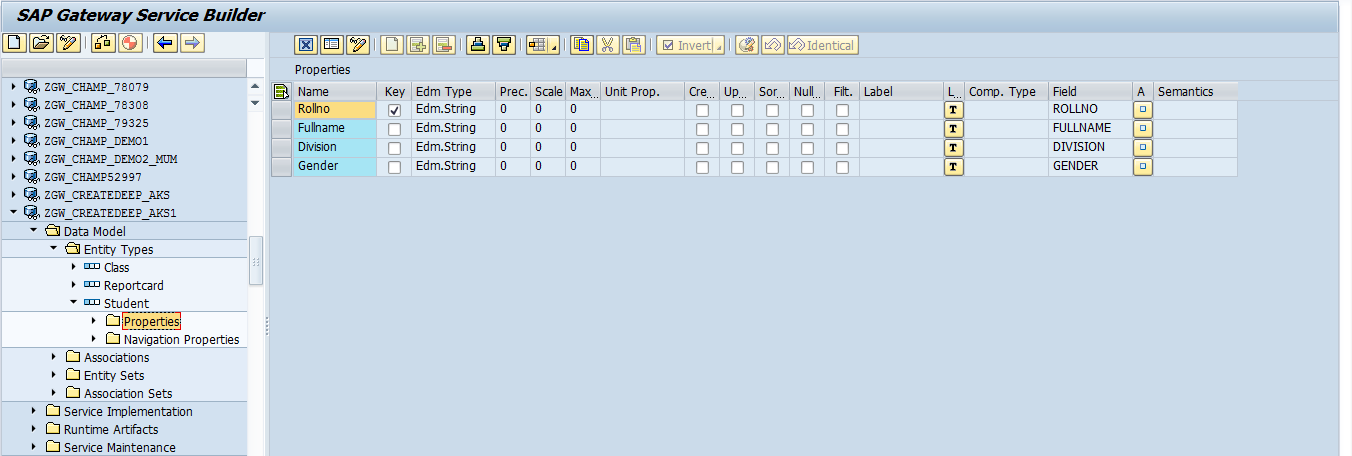
1. Class



1. Reportcard



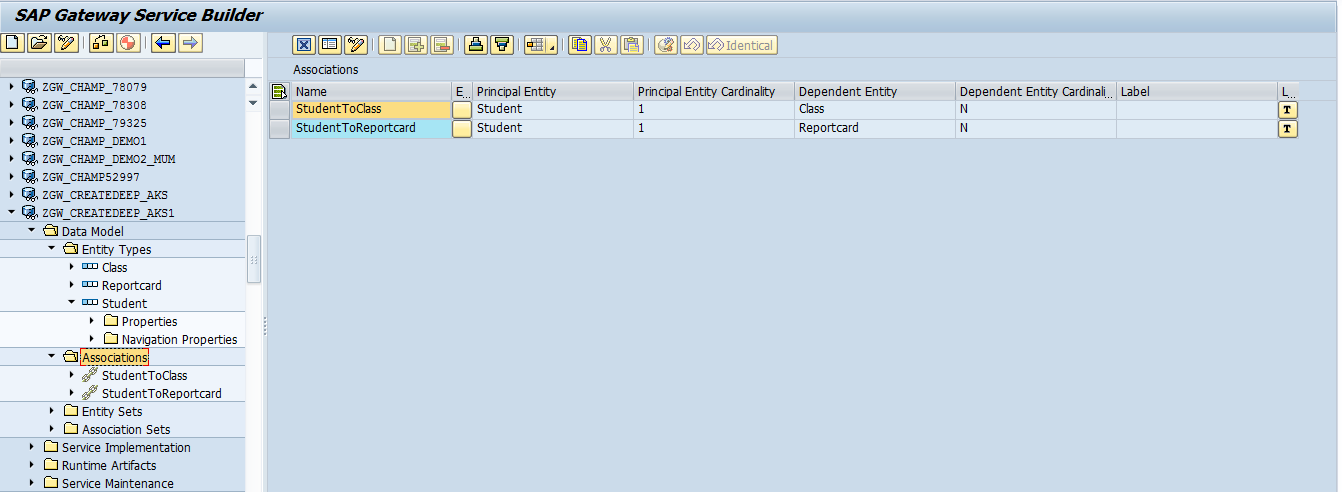
1. Student (Student is the header or parent entity)



Step 2:

Association Properties

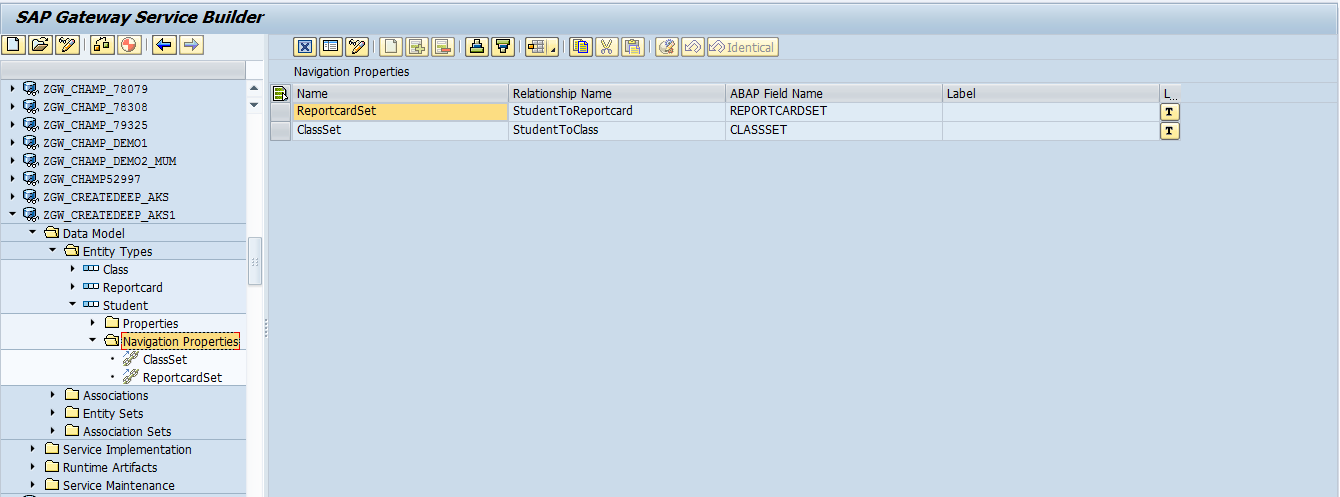
1. From Student to Class
2. From Student to Reportcard



Step 3 :

Navigation properties: (Created automatically once the association properties are defined)

1. ClassSet
2. ReportcardSet



Step 4 :

Define **TYPES** (TS\_DEEP\_ENTITY) for deep entity in ZCL\_ZGW\_CREATEDEEP\_\_01\_MPC\_EXT

CLASS zcl\_zgw\_createdeep\_\_01\_mpc\_ext DEFINITION  
  PUBLIC  
  INHERITING FROM zcl\_zgw\_createdeep\_\_01\_mpc  
  CREATE PUBLIC .  
  
  PUBLIC SECTION.  
  
    TYPES:  
      BEGIN  OF ts\_deep\_entity ,  
        rollno        TYPE string,  
        fullname      TYPE string,  
        division      TYPE string,  
        gender        TYPE string,  
        classset      TYPE STANDARD TABLE OF ts\_class WITH DEFAULT KEY,  
        reportcardset TYPE STANDARD TABLE OF ts\_reportcard WITH DEFAULT KEY,  
      END OF ts\_deep\_entity .  
  
    METHODS define  
         REDEFINITION .

Step 5 :

Redefine the method **DEFINE** in ZCL\_ZGW\_CREATEDEEP\_\_01\_MPC\_EXT

Coding :

  METHOD define.  
      super->define( ).  
  DATA:  
  lo\_annotation     TYPE REF TO /iwbep/if\_mgw\_odata\_annotation,  
  lo\_entity\_type    TYPE REF TO /iwbep/if\_mgw\_odata\_entity\_typ,  
  lo\_complex\_type   TYPE REF TO /iwbep/if\_mgw\_odata\_cmplx\_type,  
  lo\_property       TYPE REF TO /iwbep/if\_mgw\_odata\_property,  
  lo\_entity\_set     TYPE REF TO /iwbep/if\_mgw\_odata\_entity\_set.  
  
  lo\_entity\_type = model->get\_entity\_type( iv\_entity\_name = 'Student' ). "#EC NOTEXT  
  
  lo\_entity\_type->bind\_structure( iv\_structure\_name  = 'ZCL\_ZGW\_CREATEDEEP\_\_01\_MPC\_EXT=>TS\_DEEP\_ENTITY' ).   
  ENDMETHOD.

Step 6 :

Create Custom method **CUSTOM\_CREATE\_DEEP\_ENTITY in** classZCL\_ZGW\_CREATEDEEP\_\_01\_MPC\_EXT

With following parameters and exceptions

IV\_ENTITY\_NAME                          Importing            Type      STRING

IV\_ENTITY\_SET\_NAME                 Importing            Type      STRING

IV\_SOURCE\_NAME                       Importing            Type      STRING

IT\_KEY\_TAB                                   Importing            Type      /IWBEP/T\_MGW\_NAME\_VALUE\_PAIR

IT\_NAVIGATION\_PATH                Importing            Type      /IWBEP/T\_MGW\_NAVIGATION\_PATH

IO\_EXPAND                                   Importing            Type Ref To        /IWBEP/IF\_MGW\_ODATA\_EXPAND

IO\_TECH\_REQUEST\_CONTEXT    Importing            Type Ref To        /IWBEP/IF\_MGW\_REQ\_ENTITY\_C

IO\_DATA\_PROVIDER                    Importing            Type Ref To        /IWBEP/IF\_MGW\_ENTRY\_PROVIDER

ER\_DEEP\_ENTITY                           Exporting           Type      ZCL\_ZGW\_CREATEDEEP\_\_01\_MPC\_EXT =>TS\_DEEP\_ENTITY

/IWBEP/CX\_MGW\_BUSI\_EXCEPTION

/IWBEP/CX\_MGW\_TECH\_EXCEPTION

Coding:

  METHOD CUSTOM\_CREATE\_DEEP\_ENTITY.  
  
"Data Declaration  
    DATA:   lr\_deep\_entity TYPE zcl\_zgw\_createdeep\_\_01\_mpc\_ext=>ts\_deep\_entity,  
            lt\_class       TYPE zcl\_zgw\_createdeep\_\_01\_mpc\_ext=>tt\_class,  
            ls\_class       TYPE zcl\_zgw\_createdeep\_\_01\_mpc\_ext=>ts\_class,  
            lt\_reportcard  TYPE zcl\_zgw\_createdeep\_\_01\_mpc\_ext=>tt\_reportcard,  
            ls\_reportcard  TYPE zcl\_zgw\_createdeep\_\_01\_mpc\_ext=>ts\_reportcard,  
  
            lt\_istudent    TYPE TABLE OF zstudent,  
            ls\_istudent    LIKE LINE OF lt\_istudent,  
            lt\_iclass      TYPE TABLE OF zclass,  
            ls\_iclass      LIKE LINE OF lt\_iclass,  
            lt\_ireportcard TYPE TABLE OF zreportcard,  
            ls\_ireportcard LIKE LINE OF lt\_ireportcard.  
  
"Getting data from the Payload  
    io\_data\_provider->read\_entry\_data(  
    IMPORTING  
    es\_data = lr\_deep\_entity ).  
  
"Putting data into the structure and then into their respective internal tables  
"Filling up Student Internal table  
    ls\_istudent-rollno     =   lr\_deep\_entity-rollno.  
    ls\_istudent-fullname   =   lr\_deep\_entity-fullname.  
    ls\_istudent-division   =   lr\_deep\_entity-division.  
    ls\_istudent-gender     =   lr\_deep\_entity-gender.  
    APPEND ls\_istudent TO lt\_istudent.  
  
"Filling up Class Internal table  
    FIELD-SYMBOLS: <ls\_class> TYPE zcl\_zgw\_createdeep\_\_01\_mpc\_ext=>ts\_class.  
  
    LOOP AT lr\_deep\_entity-classset ASSIGNING <ls\_class>.  
      CLEAR  ls\_iclass.  
      ls\_iclass-rollno          = <ls\_class>-rollno.  
      ls\_iclass-fullname        = <ls\_class>-fullname.  
      ls\_iclass-std             = <ls\_class>-std.  
  
      APPEND  ls\_iclass TO lt\_iclass.  
      CLEAR: <ls\_class>,ls\_iclass.  
    ENDLOOP.  
  
"Filling up Reportcard Internal table  
    FIELD-SYMBOLS: <ls\_reportcard> TYPE zcl\_zgw\_createdeep\_\_01\_mpc\_ext=>ts\_reportcard.  
  
    LOOP AT lr\_deep\_entity-reportcardset ASSIGNING <ls\_reportcard>.  
      CLEAR  ls\_ireportcard.  
      ls\_ireportcard-rollno          = <ls\_reportcard>-rollno.  
      ls\_ireportcard-fullname        = <ls\_reportcard>-fullname.  
      ls\_ireportcard-maths           = <ls\_reportcard>-maths.  
      ls\_ireportcard-science         = <ls\_reportcard>-science.  
      ls\_ireportcard-total           = <ls\_reportcard>-total.  
      ls\_ireportcard-grade           = <ls\_reportcard>-grade.  
      ls\_ireportcard-remarks         = <ls\_reportcard>-remarks.  
  
      APPEND  ls\_ireportcard TO lt\_ireportcard.  
      CLEAR: <ls\_reportcard>,ls\_ireportcard.  
    ENDLOOP.  
  
    INSERT zstudent     FROM TABLE lt\_istudent.  
    INSERT zclass       FROM TABLE lt\_iclass.  
    INSERT zreportcard  FROM TABLE lt\_ireportcard.  
"Or Any other functionality needed such as calling the Function Module  
  ENDMETHOD.

Step 7 :

Redefine the method /IWBEP/IF\_MGW\_APPL\_SRV\_RUNTIME~CREATE\_DEEP\_ENTITY

  METHOD /iwbep/if\_mgw\_appl\_srv\_runtime~create\_deep\_entity.  
  
    DATA: custom\_create\_deep\_entity TYPE ZCL\_ZGW\_CREATEDEEP\_\_01\_mPC\_EXT=>ts\_deep\_entity.  
  
    CASE iv\_entity\_set\_name.  
\*————————————————————————-\*  
\*             EntitySet –  HeaderSet  
\*————————————————————————-\*  
      WHEN 'StudentSet'.  
\*       Call the entity set generated method  
  
        CALL METHOD me->custom\_create\_deep\_entity  
          EXPORTING  
            iv\_entity\_name          = iv\_entity\_name  
            iv\_entity\_set\_name      = iv\_entity\_set\_name  
            iv\_source\_name          = iv\_source\_name  
            it\_key\_tab              = it\_key\_tab  
            it\_navigation\_path      = it\_navigation\_path  
            io\_expand               = io\_expand  
            io\_tech\_request\_context = io\_tech\_request\_context  
            io\_data\_provider        = io\_data\_provider  
          IMPORTING  
            er\_deep\_entity          = custom\_create\_deep\_entity.  
  
        copy\_data\_to\_ref(  
        EXPORTING  
        is\_data = custom\_create\_deep\_entity  
        CHANGING  
        cr\_data = er\_deep\_entity  
        ).  
    ENDCASE.  
  ENDMETHOD.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Coding Part is Done\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

GW Client

1. Using POST method with the following URI

/sap/opu/odata/sap/ZGW\_CREATEDEEP\_AKS1\_SRV/StudentSet

1. Use the payload mentioned below

