**ASSIGNMENT (DAY 10)**

**Name: Abantika Das Modak**

**Employee ID: 46247685**

**Package: ZABN\_PKG\_LIT**

**Assignment 3 – Business Add in (BADI)-Single Use**

**Hint:**

1. **SE18 should be used to define a BADI**
2. **SE19 should be used to implement the BADI Definition.**

**Requirement:** Define and implement a custom BADI in SE18 to divide two numbers.

Step1: Define a BADI (ZBADI\_DEMO\_01…05) in SE18.

Step2: Double click on the interface created to define the methods, parameters and exceptions.

Step3: Define an instance method DIVIDE\_NUMBERS.

Step4: Define two importing parameter’s (I\_NUM1 & I\_NUM2) and one exporting parameter (E\_RESULT) of type integer

Step5: Define an exception if division is carried out using null value.

Step6: Create an implementation (ZBADI\_DEMO\_IMP\_01...05) by specifying the BADI name that was created in step1.u

Step7: Double click on the method and write the source code to divide the two numbers.

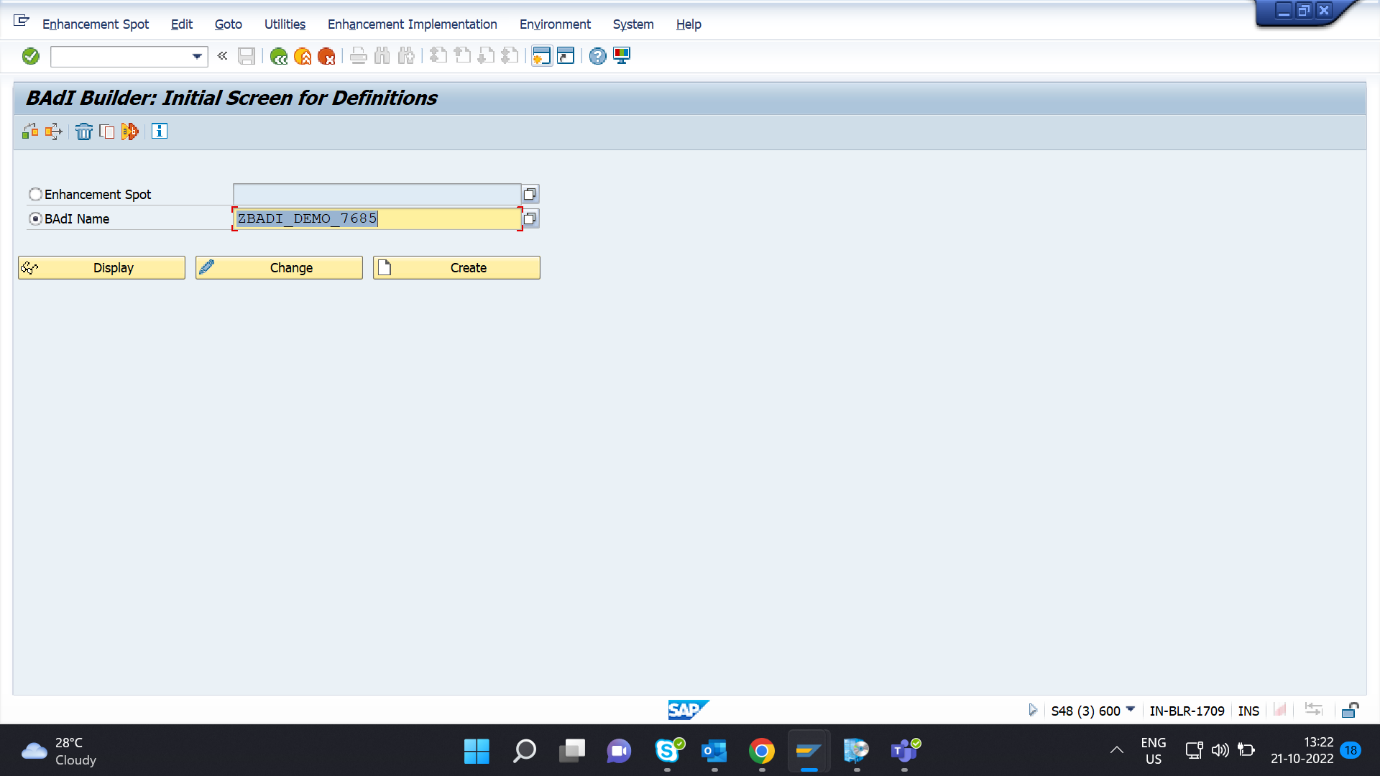
Step8: Create a wrapper program (ZDEMO\_BADI\_01…05).

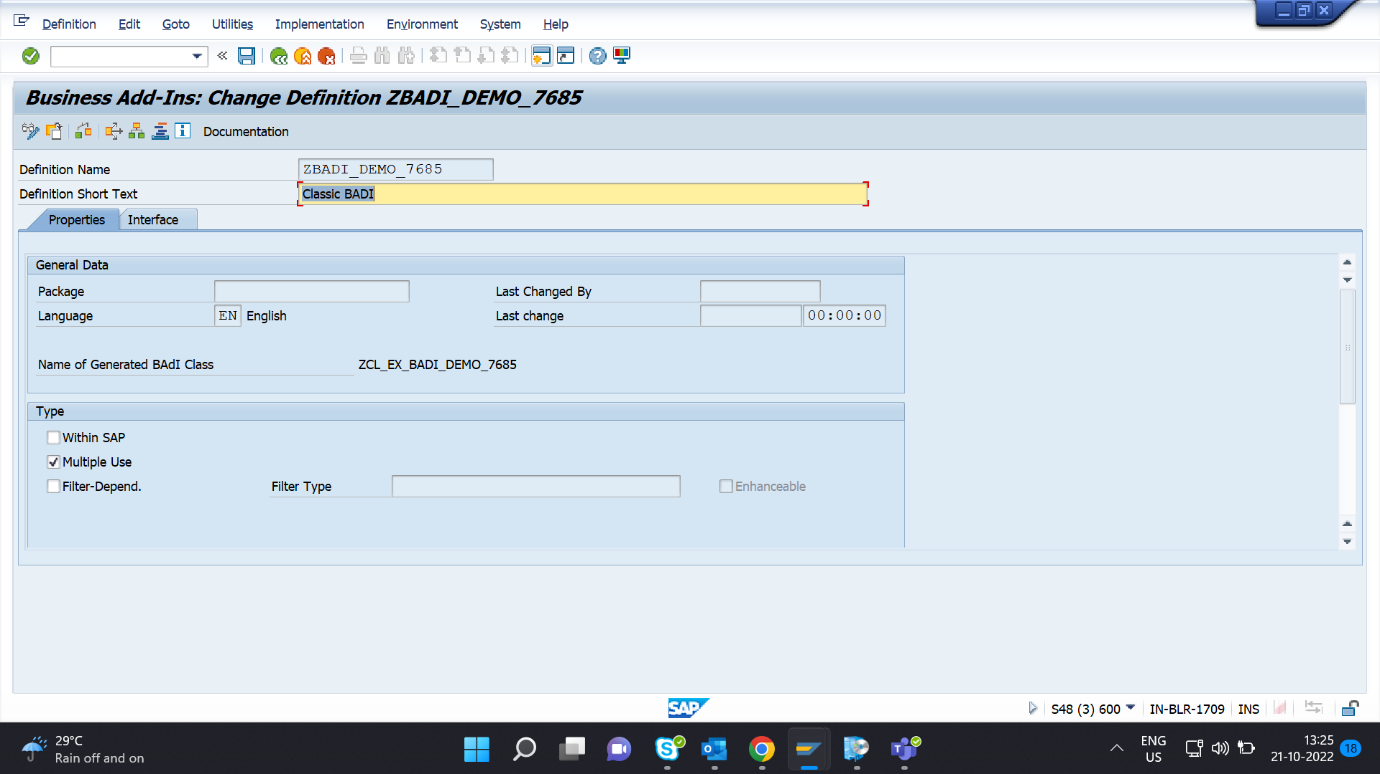
Step9: Define a reference variable of type BADI interface.

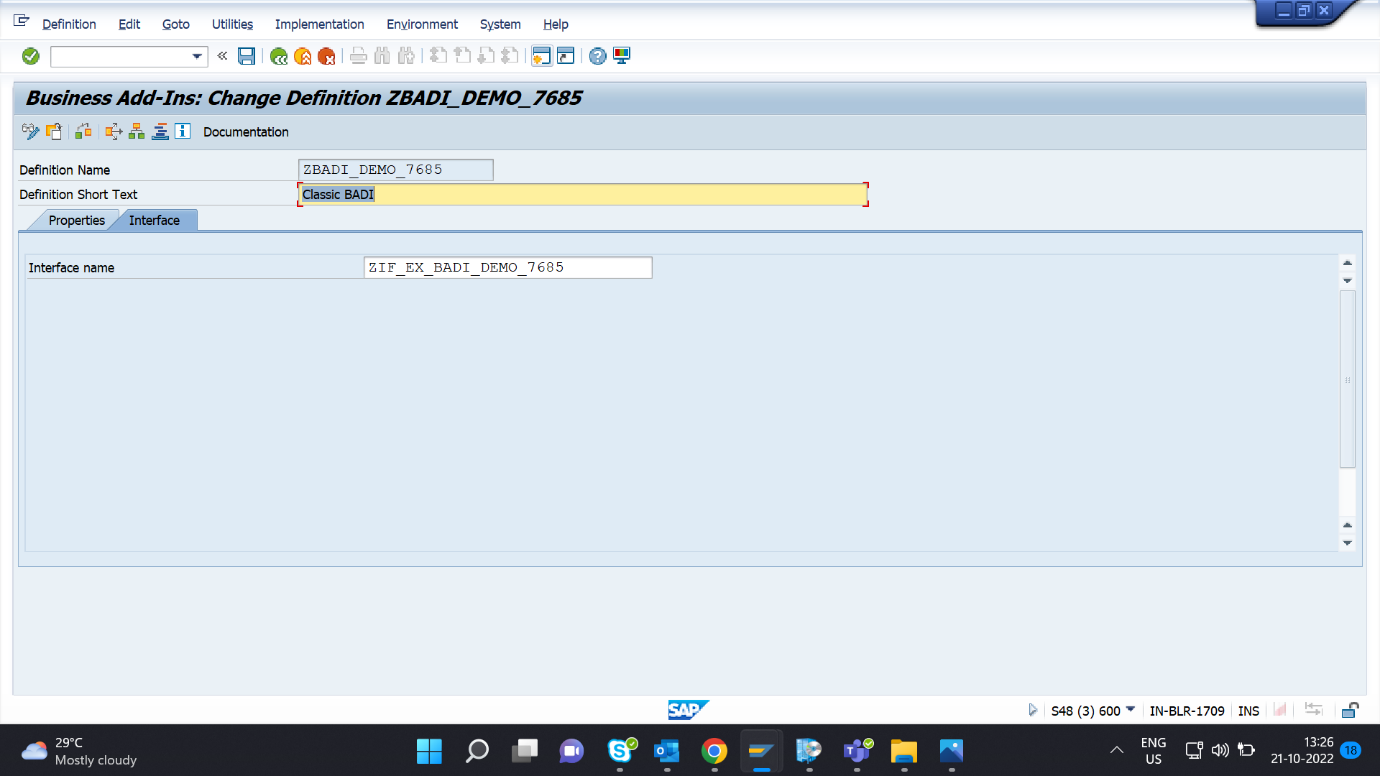
Step10: Define selection screen with two parameters of type integers

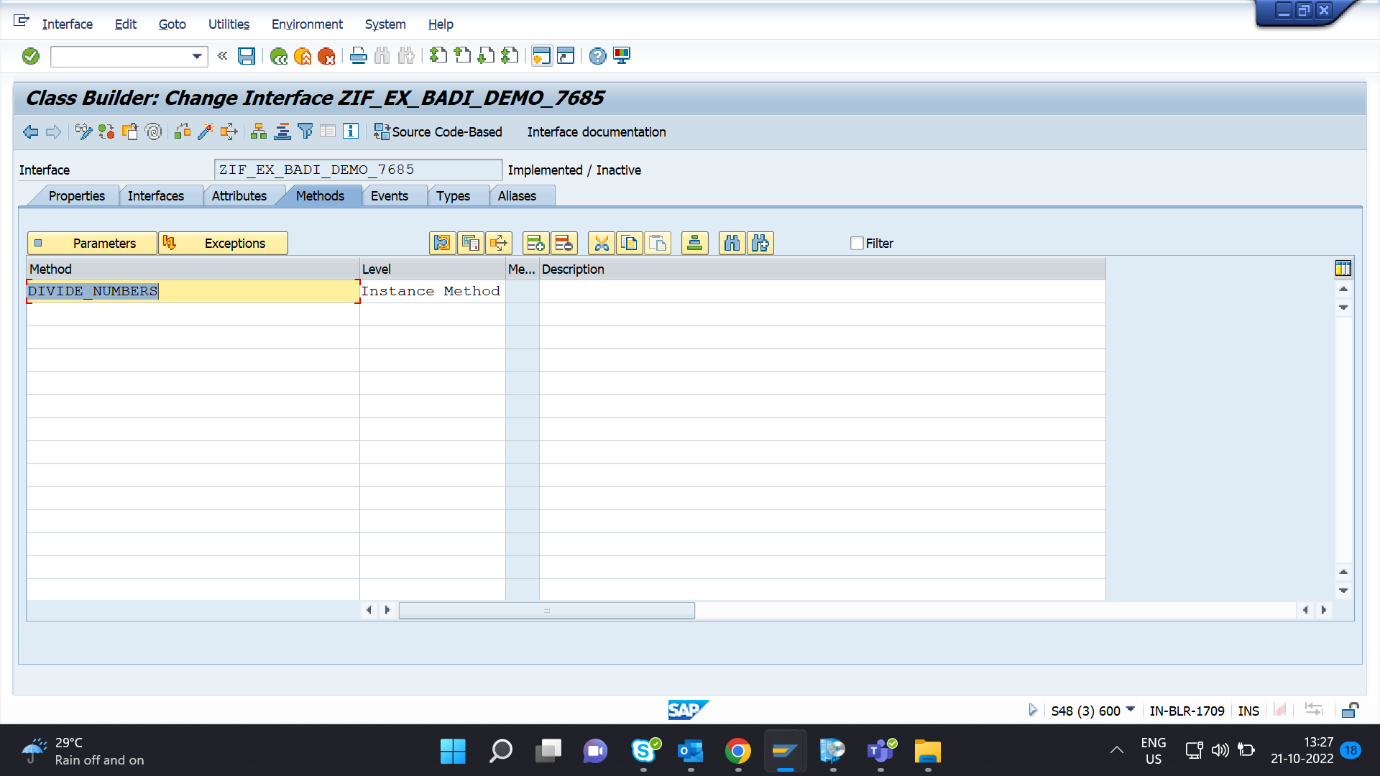
Step11: Call method GET\_INSTANCE of the class CL\_EXITHANDLER to get the BADI instance

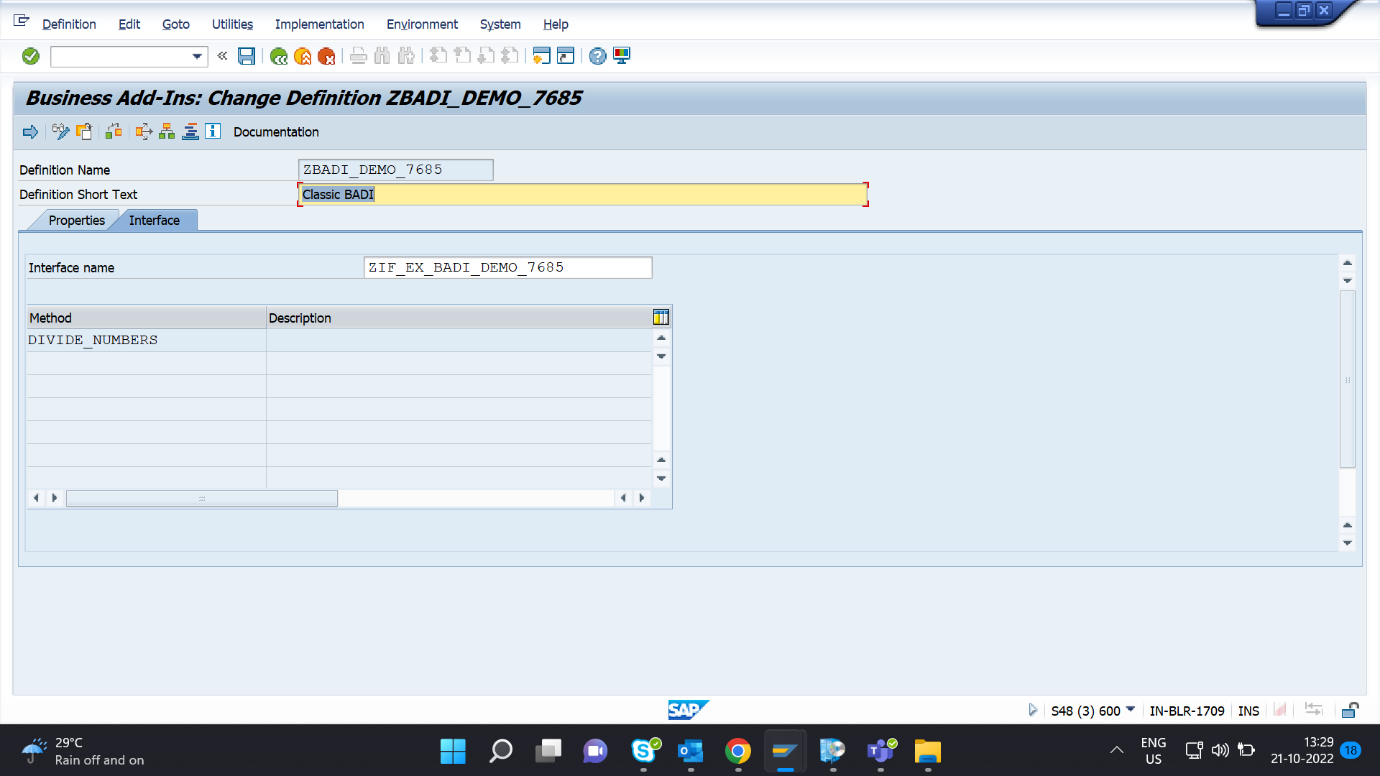
Step12: Call method DIVIDE\_NUMBERS using this instance and pass the two selection parameters.

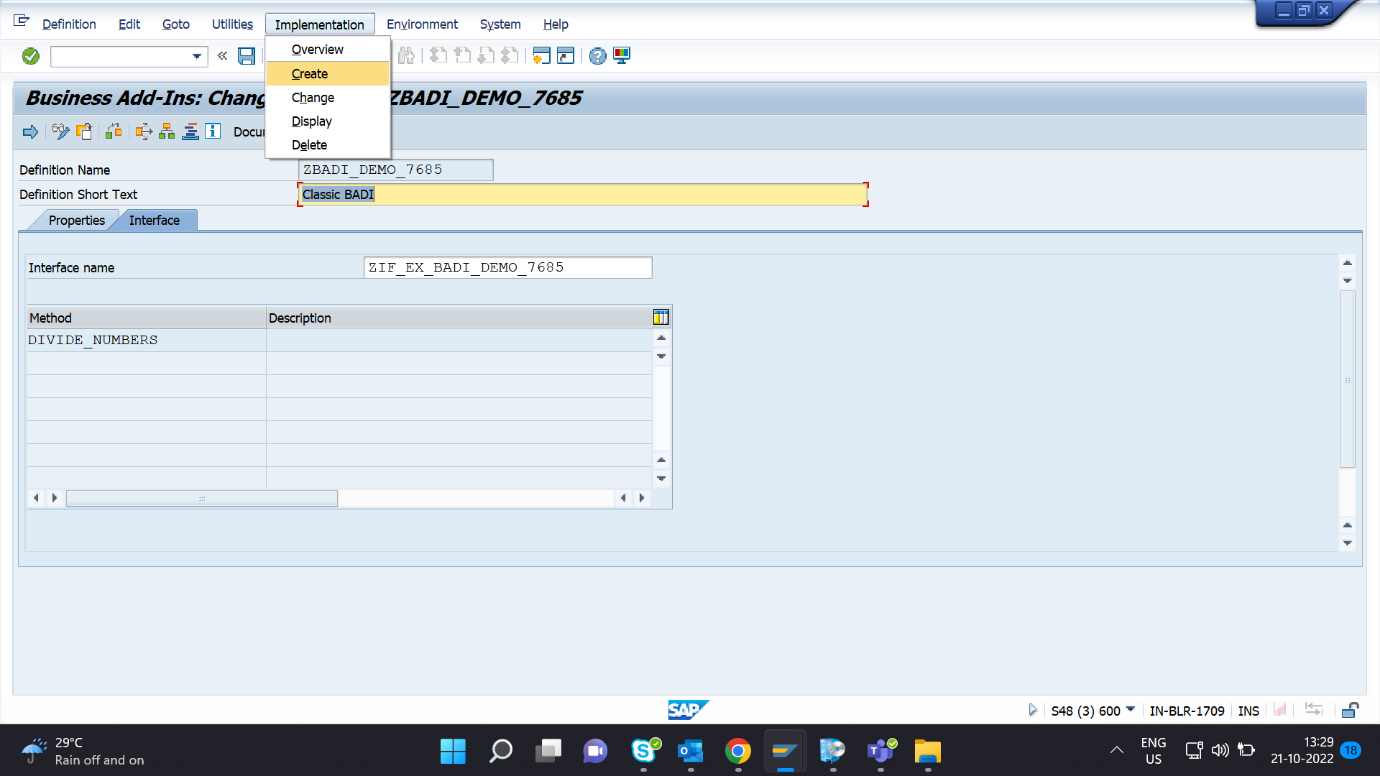


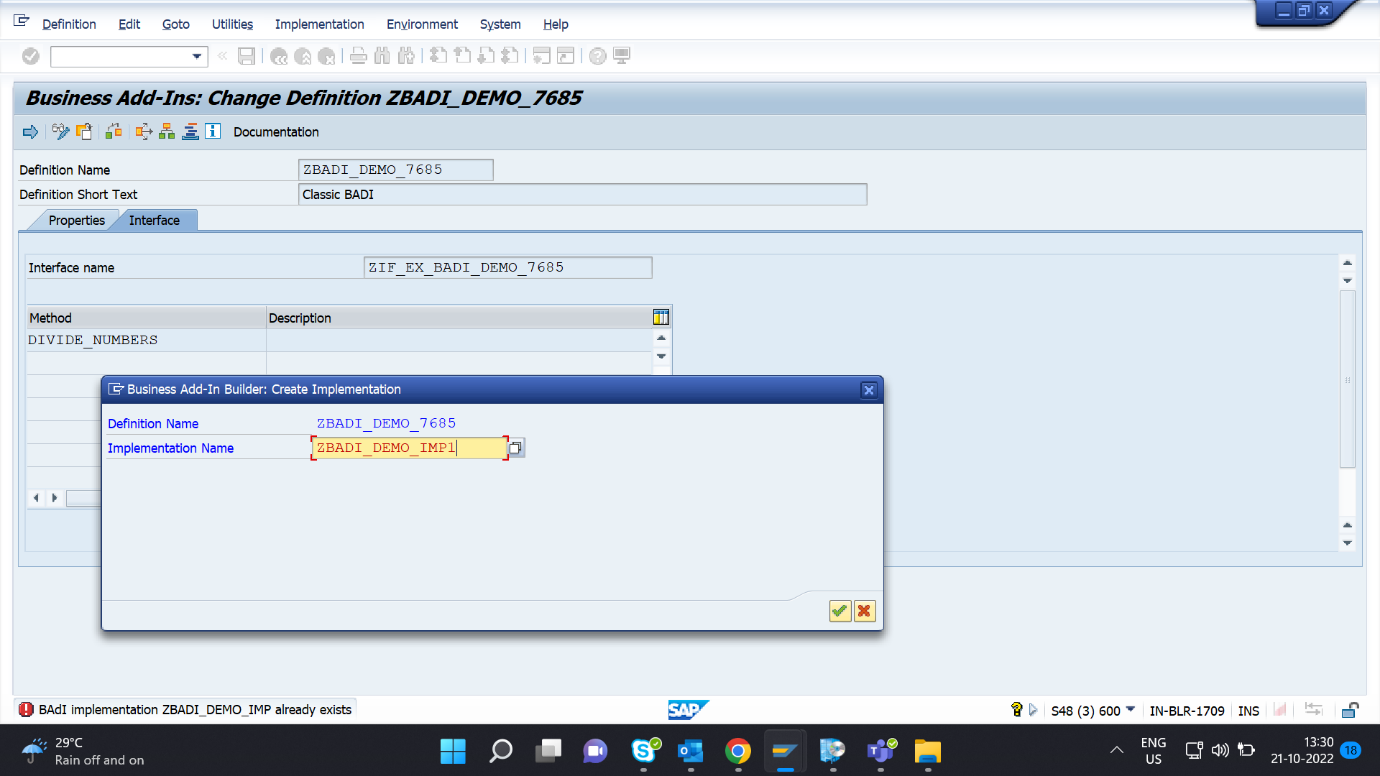


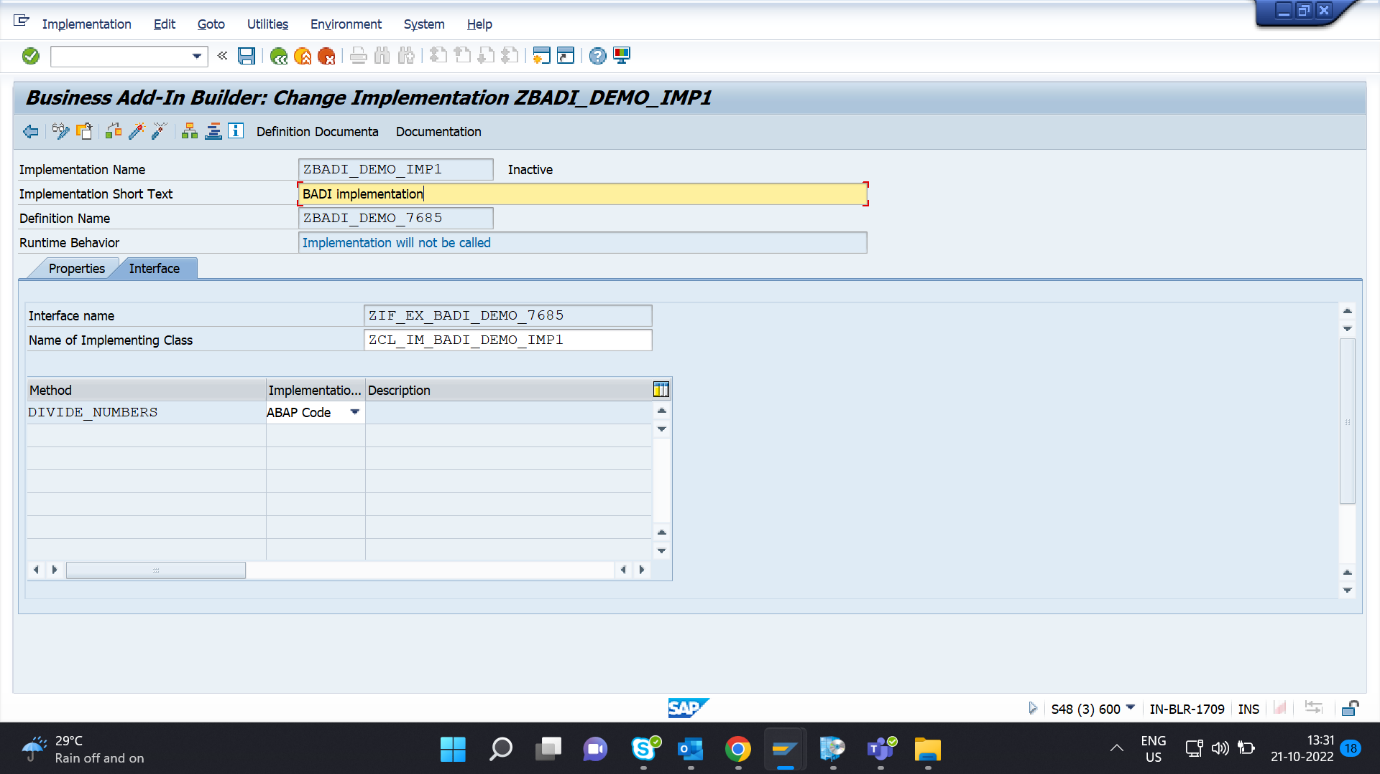




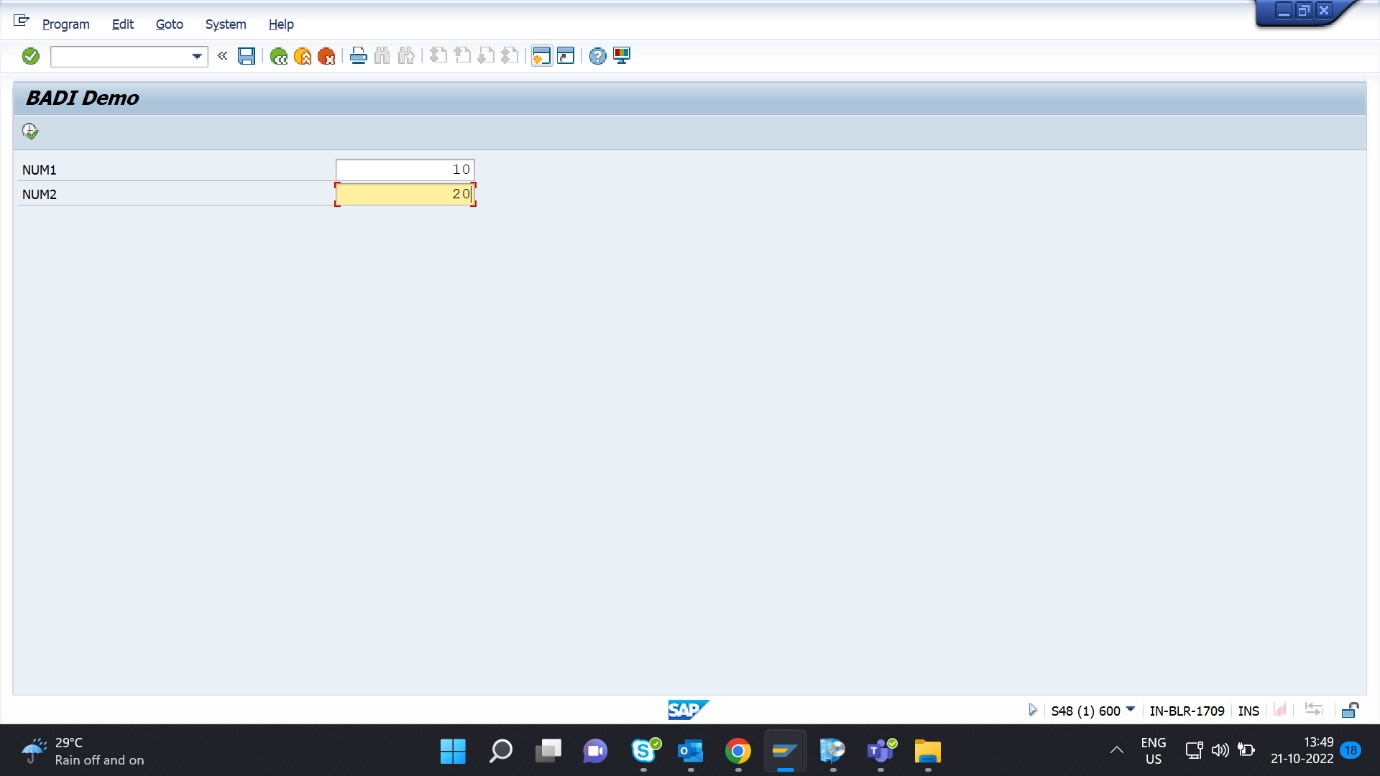


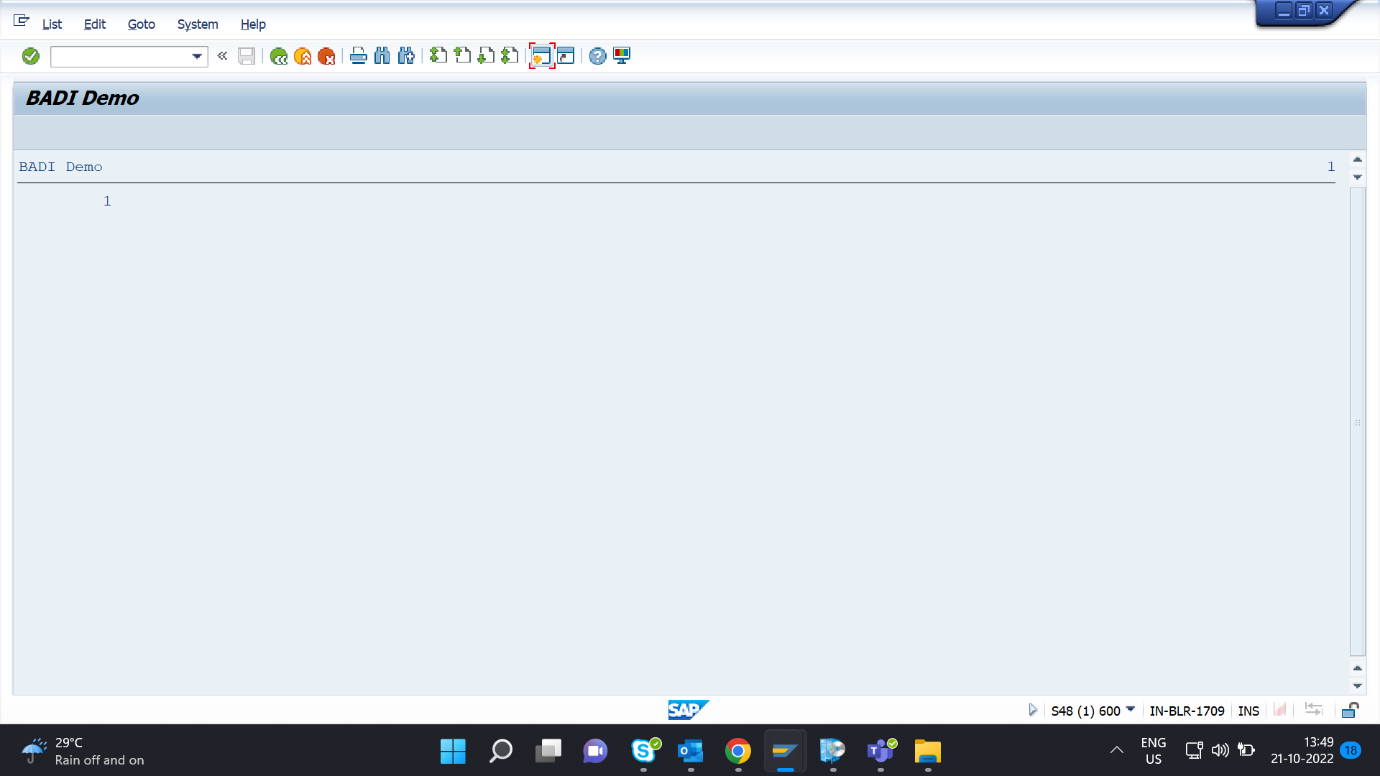






REPORT Z7685\_BADI\_DEMO.  
  
START-OF-SELECTION. CLASS CL\_EXITHANDLER DEFINITION LOAD.  
DATA: RESULT TYPE I.  
DATA OBJ1 TYPE REF TO ZIF\_EX\_BADI\_DEMO\_7685.  
  
PARAMETERS: NUM1 TYPE I,  
            NUM2 TYPE I.  
  
CALL METHOD cl\_exithandler=>get\_instance  
  EXPORTING  
    exit\_name                     =  'ZBADI\_DEMO\_7685'  
\*    null\_instance\_accepted        = SEEX\_FALSE  
\*  IMPORTING  
\*    act\_imp\_existing              =  
  CHANGING  
    instance                      = OBJ1  
  EXCEPTIONS  
    no\_reference                  = 1  
    no\_interface\_reference        = 2  
    no\_exit\_interface             = 3  
    class\_not\_implement\_interface = 4  
    single\_exit\_multiply\_active   = 5  
    cast\_error                    = 6  
    exit\_not\_existing             = 7  
    data\_incons\_in\_exit\_managem   = 8  
    others                        = 9  
        .  
IF sy-subrc <> 0.  
\* Implement suitable error handling here  
ENDIF.  
  
TRY.  
  CALL METHOD obj1->divide\_numbers  
      EXPORTING i\_num1 = NUM1  
                i\_num2 = NUM2 .  
 CATCH cx\_sy\_zerodivide .  
  
   ENDTRY.





**Assignment 4 – Business Add in (BADI)-Multiple Use**

**Hint:**

1. **SE18 should be used to define a BADI**
2. **SE19 should be used to implement the BADI Definition.**

**Requirement:** Define and implement a custom BADI in SE18 to get Booking Data & Emailid.

Step1: Define a BADI (ZBADI\_DEMO\_01…05) in SE18.

Step2: Double click on the interface created to define the methods, parameters and exceptions.

Step3: Input Carrid & Connid Table SCARR Get, Email Data.

Step4: Input Carrid & Connid Table SBOOK Get Booking Data.

Step5: Define an exception if no data is present.

Step6: Create an implementation (ZBADI\_DEMO\_IMP\_01...05) by specifying the BADI name that was created in step1.

Step7: Create a wrapper program (ZDEMO\_BADI\_01…05).

Step8: Define a reference variable of type BADI interface.

Step9: Define selection screen with two parameters of type integers

Step10: Call method GET\_INSTANCE of the class CL\_EXITHANDLER to get the BADI instance

Step11: Call method DIVIDE\_NUMBERS using this instance and pass the two selection parameters.

Create a program to display Data of Sflight

CARRID

CONNID

FLDATE

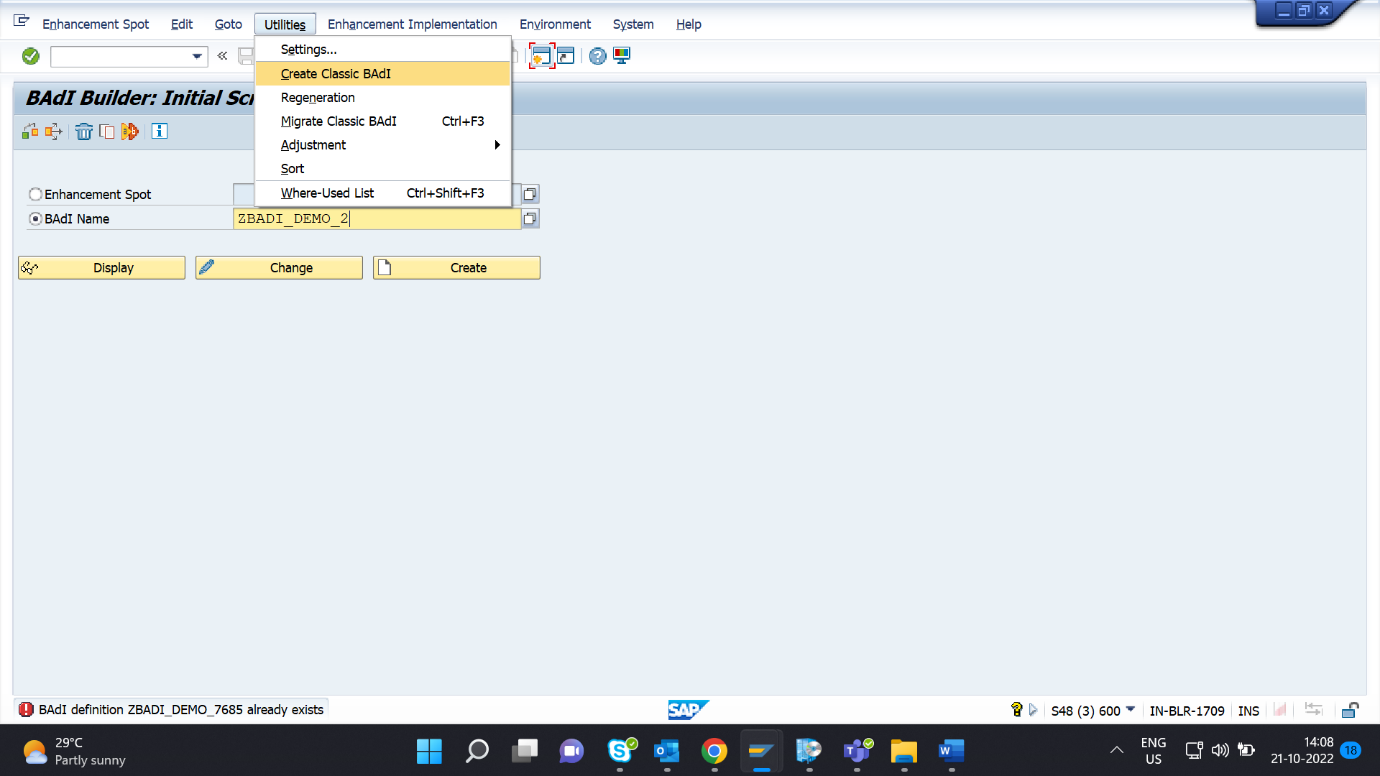
PRICE

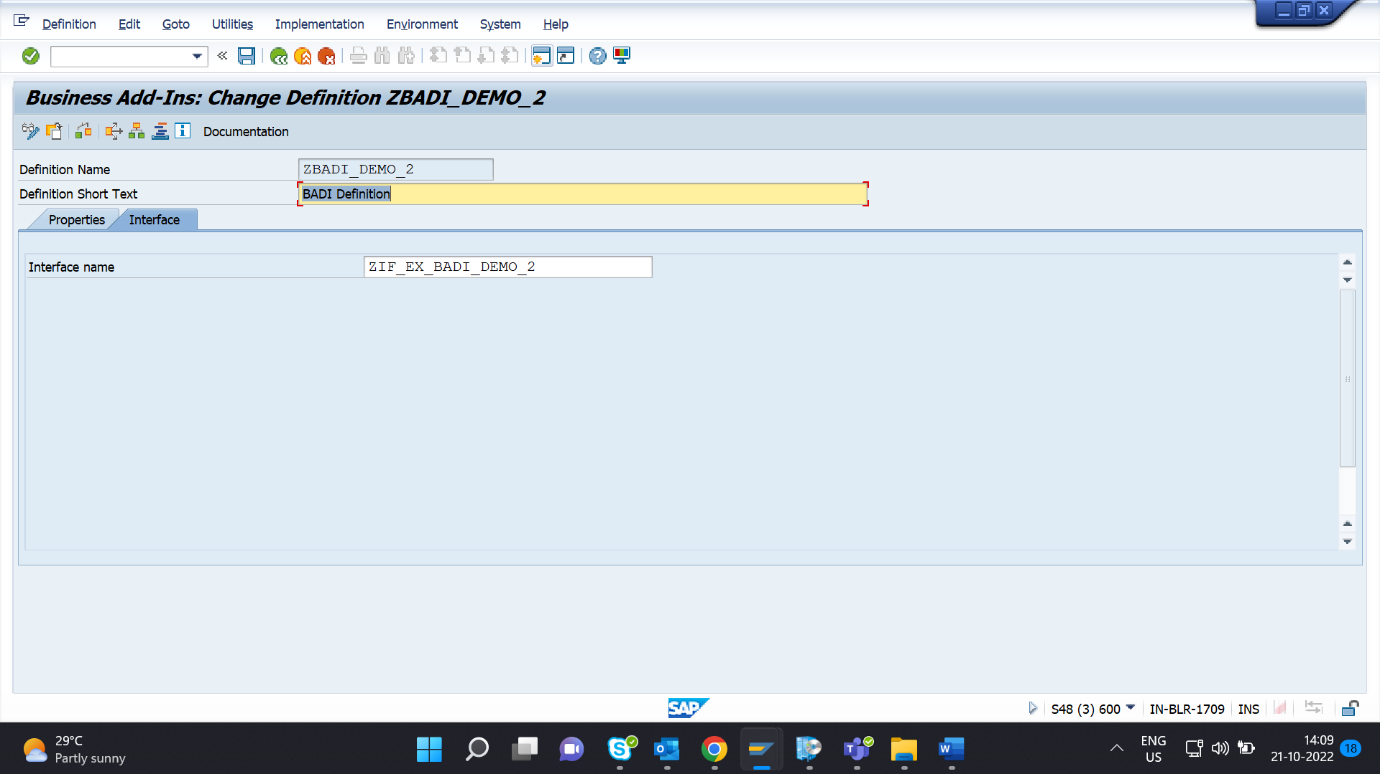
CURRENCY

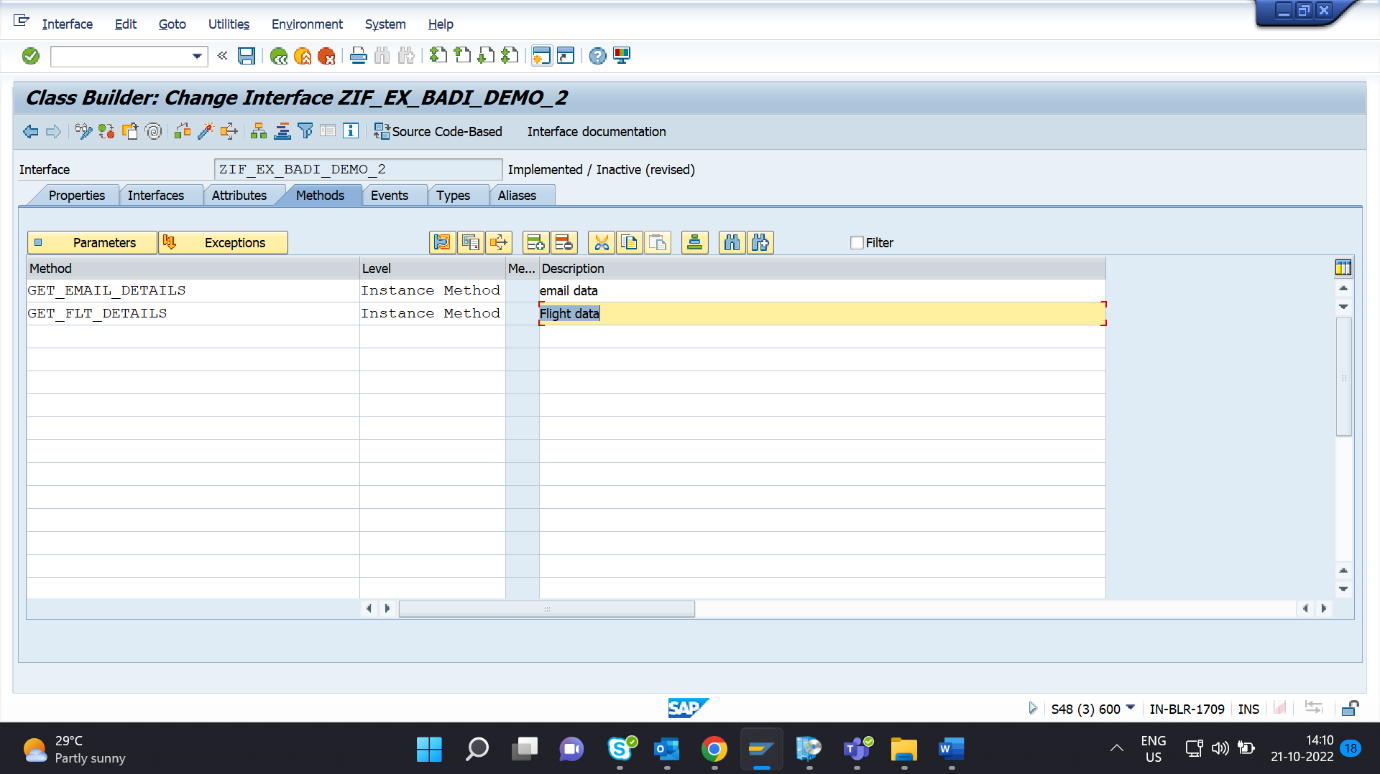
PLANETYPE

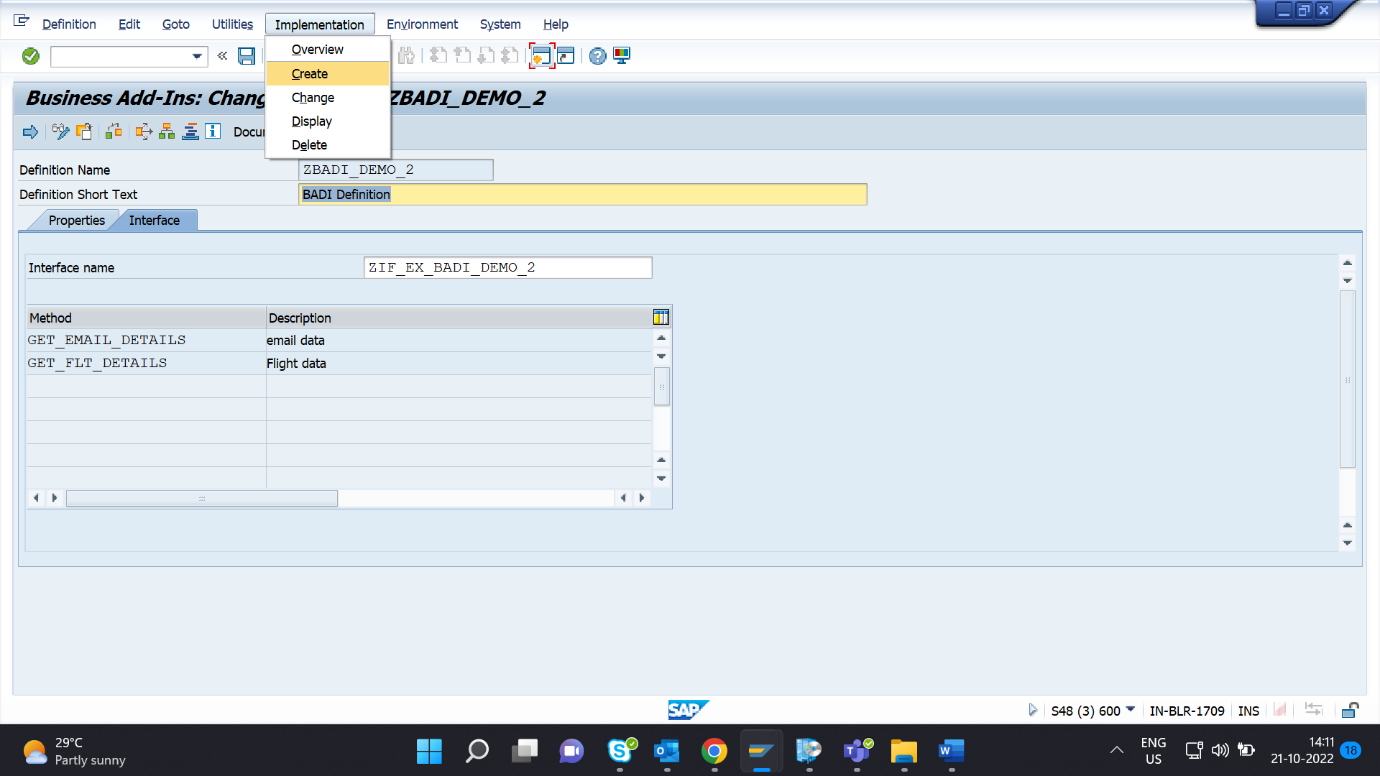
SEATSMAX

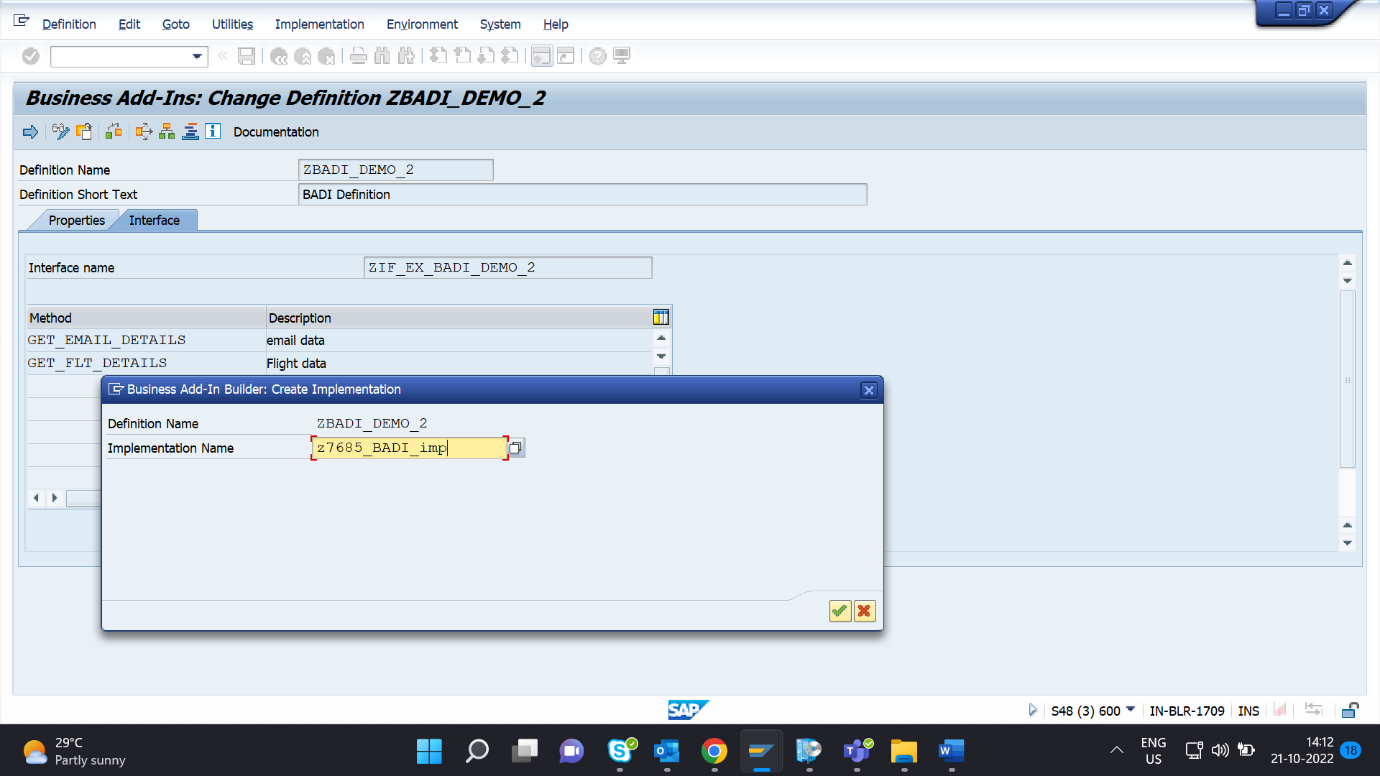
**URL-SCARR---(From Badi)**

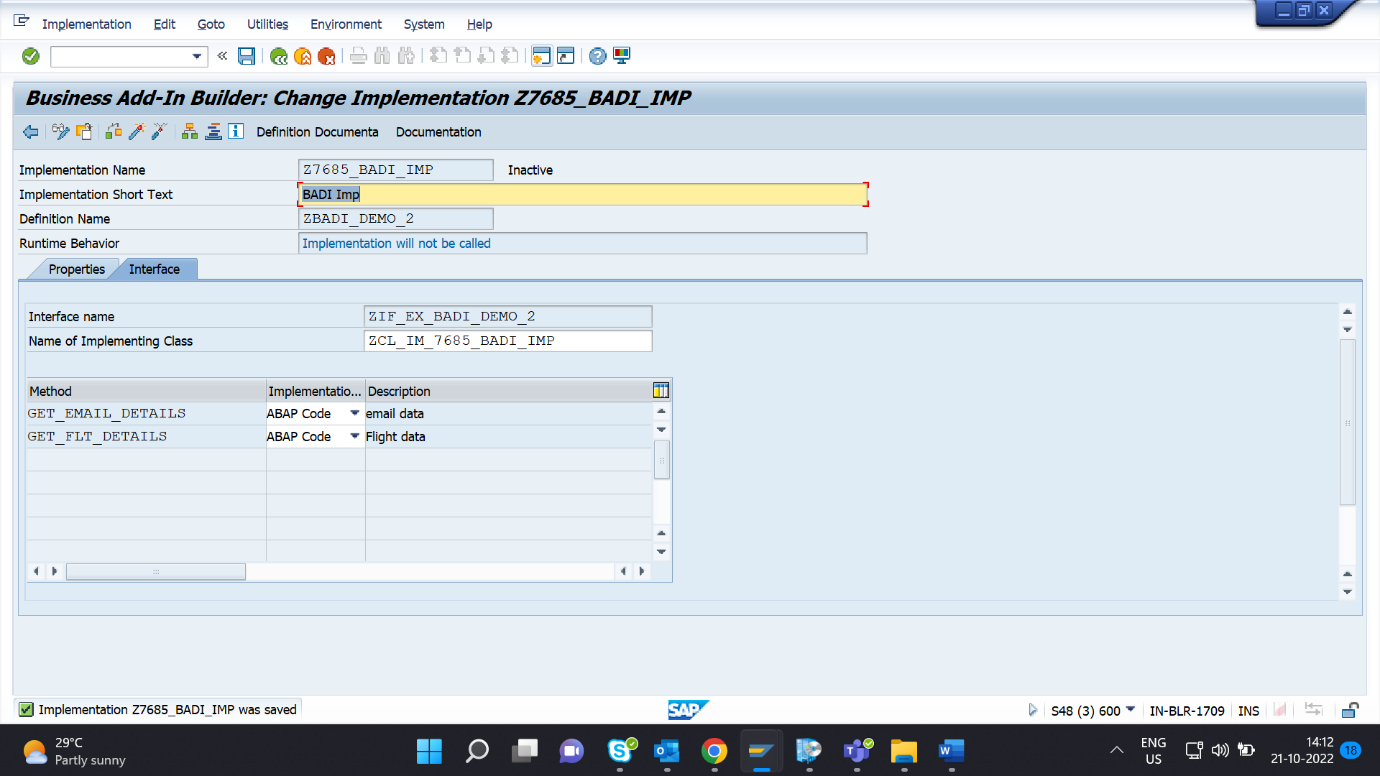


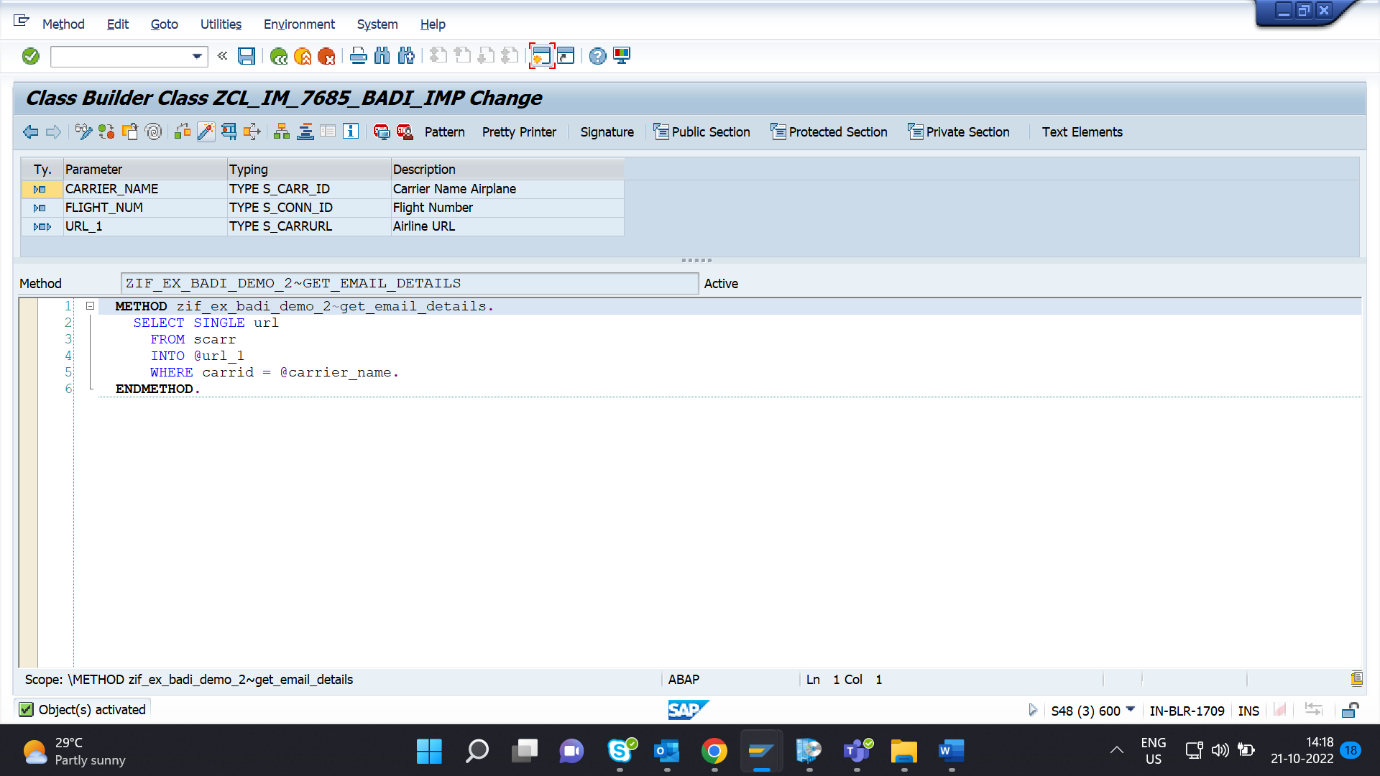


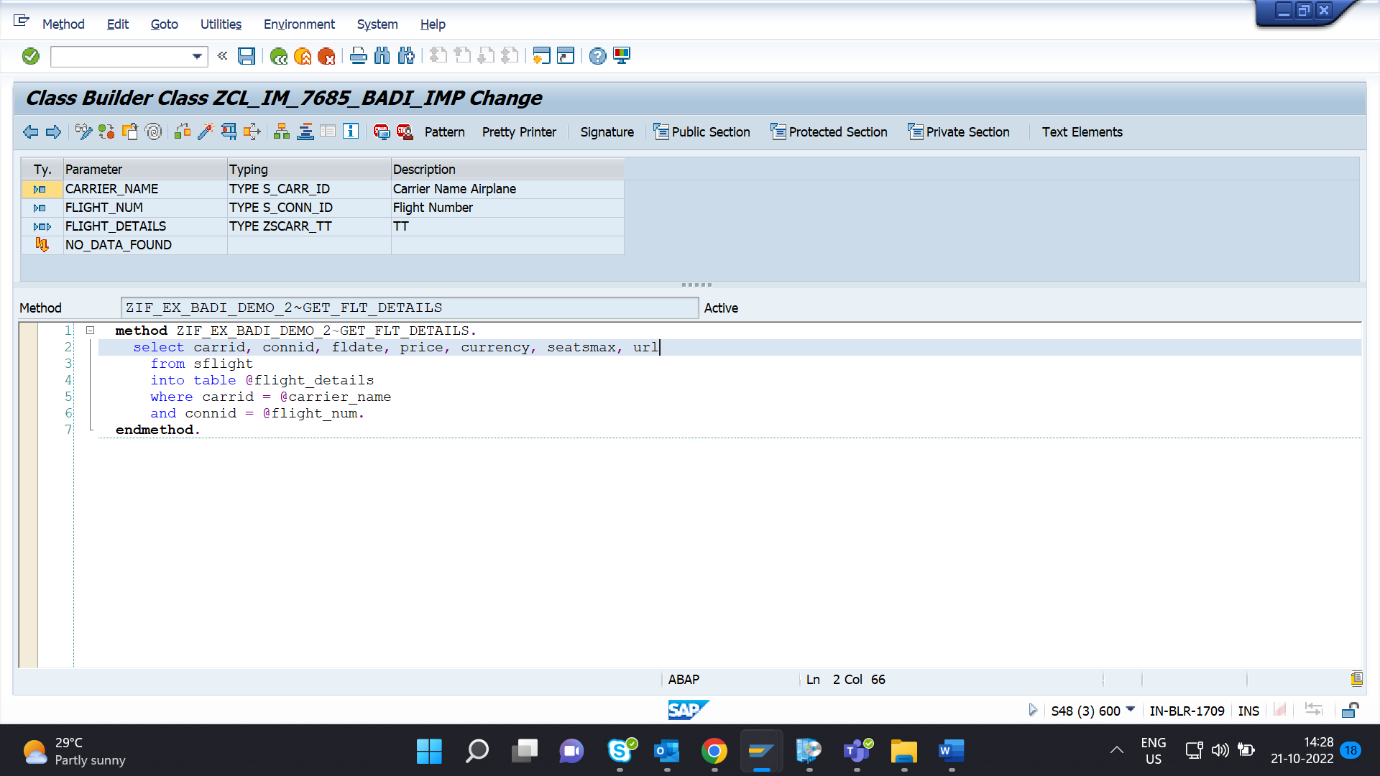












REPORT ZABN\_DAY10\_Q4.  
  
DATA: OBJ TYPE REF TO ZIF\_EX\_BADI\_D10\_LIT.  
  
PARAMETERS: p\_carr TYPE S\_CARR\_ID,  
            P\_flight TYPE S\_CONN\_ID.  
  
  
DATA : it\_flight TYPE  Zscarr\_tt,                          " declaring internal table of sflight  
       gs\_sflight TYPE line of Zscarr\_tt,  
       url type s\_carrurl.  
CL\_EXITHANDLER=>GET\_INSTANCE(  
EXPORTING  
    EXIT\_NAME                     =  'ZBADI\_D10\_LIT'   " Business Add-In Definition  
\*    NULL\_INSTANCE\_ACCEPTED        =     " Is Null Instance OK if There Is no Active Implementation?  
\*  IMPORTING  
\*    ACT\_IMP\_EXISTING              =     " Returns an Active Implementation  
CHANGING  
    INSTANCE                      =  OBJ   " Instance  
EXCEPTIONS  
    NO\_REFERENCE                  = 1  
    NO\_INTERFACE\_REFERENCE        = 2  
    NO\_EXIT\_INTERFACE             = 3  
    CLASS\_NOT\_IMPLEMENT\_INTERFACE = 4  
    SINGLE\_EXIT\_MULTIPLY\_ACTIVE   = 5  
    CAST\_ERROR                    = 6  
    EXIT\_NOT\_EXISTING             = 7  
    DATA\_INCONS\_IN\_EXIT\_MANAGEM   = 8  
OTHERS                        = 9  
).  
IF SY-SUBRC <> 0.  
\* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO  
\*            WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.  
ENDIF.  
  
CALL METHOD obj->get\_flt\_details  
  EXPORTING  
    carrier\_name   = p\_carr  
    flight\_num     = p\_flight  
  CHANGING  
    flight\_details = it\_flight  
\*  EXCEPTIONS  
\*    no\_data\_found  = 1  
\*    others         = 2  
        .  
IF sy-subrc <> 0.  
\* Implement suitable error handling here  
ENDIF.  
  
  
  
  
  
CALL METHOD obj->get\_email\_details  
  EXPORTING  
    carrier\_name = p\_carr  
    flight\_num   = p\_flight  
  CHANGING  
    url          = url  
    .  
  
  
  
WRITE: / 'Airline URL is' COLOR COL\_POSITIVE,  url.  
WRITE: 40 'Flight Details' COLOR COL\_POSITIVE.  
WRITE: / 'CARRID',  
10 'CONNID',  
20 'FLDATE',  
31 '      PRICE',  
52 'CURRENCY',  
63 'PLANETYPE',  
75 'SEATSMAX'.  
ULINE.  
LOOP AT it\_flight INTO gs\_sflight.  
WRITE : /  gs\_sflight-carrid LEFT-JUSTIFIED,  
10 gs\_sflight-connid,  
20 gs\_sflight-fldate,  
31 gs\_sflight-price,  
52 gs\_sflight-currency,  
63 gs\_sflight-planetype,  
75 gs\_sflight-seatsmax.  
ENDLOOP.

