&----------------------------------------------------------------------------\*  
\*& REPORT ZGRP4\_EXCEL\_ALL\_TEST                                                \*  
\*&----------------------------------------------------------------------------\*  
\*& Developer Name: 280355-CHITTURI HEMALATHA                                 \*  
\*&                 280353-MALAYALA SAI SRAVAN                                 \*  
\*&                 280357-MEER ARFAT ALI                                      \*  
\*&                                                                            \*  
\*& FS Object Id: Group4 - Dynamic Table Update via Excel File Upload          \*  
\*&                                                                            \*  
\*& Object Description: -Updating/creating new entries in Database Table based \*  
\*&                      on excel file uploaded, -Getting the database table   \*  
\*&                       structure/template into excel format                 \*  
\*&                                                                            \*  
\*&\*& Transport Request: S4HK907314.                                           \*  
\*&                                                                            \*  
\*& Object Creation Date: 16.05.2024                                           \*  
\*&                                                                            \*  
\*& Functional Moudle : Any.                                                   \*  
\*&----------------------------------------------------------------------------\*  
  
REPORT ZGRP4\_EXCEL\_ALV\_FINAL.  
  
  
\*-------------------------------------------------\*  
\*  TOP INCLUDE FOR THE DECLERATIONS  
\*-------------------------------------------------\*  
INCLUDE ZGRP4\_EXCEL\_ALV\_FINALTOP.  
  
  
\*&---------------------------------------------------------------------\*  
\*&  Selection Screen Design and Processing                             \*  
\*&---------------------------------------------------------------------\*  
INCLUDE ZGRP4\_SELECTIONSCR\_IP1\_FINAL.  
  
  
\*-------------------------------------------------\*  
\*  CODE FOR DOWNLOAD TEMPLATE PUSHBUTTON  
\*-------------------------------------------------\*  
INCLUDE ZGRP4\_INPUT\_TEMPLATE1\_FINAL.  
  
  
\*-------------------------------------------------\*  
\*  Main Business logic  
\*-------------------------------------------------\*  
INCLUDE ZGRP4\_EXCEL\_TO\_ITAB1\_FINAL.  
  
  
\*-------------------------------------------------\*  
\*   UPDATE DABABASE RECORDS USING INTERNAL TABLE  
\*-------------------------------------------------\*  
INCLUDE ZGRP4\_UPLOAD\_TO\_DB1\_FINAL.  
  
  
\*-------------------------------------------------\*  
\*   Display the UPDATED RECORDS IN ooalv  
\*-------------------------------------------------\*  
INCLUDE ZGRP4\_DISPLAY\_STATUS1\_FINAL.  
  
  
\*-------------------------------------------------\*  
\*   INCLUDE PROGRAM FOR OOALV  
\*-------------------------------------------------\*  
INCLUDE ZGRP4\_DIS\_STATUS111.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Include code snippets\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*&---------------------------------------------------------------------\*  
\*& Include          ZGRP4\_EXCEL\_ALL\_TEST\_TOP  
\*&---------------------------------------------------------------------\*  
tables : dd02l. " TABLE OF DDO2L.  
TYPES : BEGIN OF ty\_tab, "structure of input template in excel file  
         fields TYPE dfies-fieldname,     "Table field names in database.  
         size   TYPE dfies-leng,    "input length permissible for the table field.  
         dtype  TYPE dfies-datatype,    "Datatype suitable for the table field.  
        END OF ty\_tab.  
  
DATA: lt\_fields      TYPE ddfields,   "TABLE TYPE FOR DETAILS OF DATABASE TABLE FIELDS  
      wa\_tab         TYPE dfies,    "Structure/work-area of selected database table row  
      lt\_headings    TYPE TABLE OF ty\_tab,    "internal table to store table field's details  
      wa\_headings    TYPE ty\_tab,   "structure upon db table headings stored in internal table  
      lv\_tabname(30) TYPE c,    "character format copy of parameter input for table name  
  
      lv\_path        TYPE string VALUE 'C:\Users\Administrator\Desktop\',     "DESKTOP PATH  
  
      lv\_fullpath    TYPE string,     "FULL PATH TO STORE EXCEL FILE IN LOCAL MACHINE.  
      lv\_xtnsn       TYPE string VALUE '\_TEMPLATE.XLSX',    "FILE EXTENSION  
      varkey         LIKE rstable-varkey,  "To store system mandate for lock tables while modifying  
  
\*&---------------------------------------------------------------------\*  
\*&Declarations for dynamic data                                        \*  
\*&---------------------------------------------------------------------\*  
      gt\_data        TYPE REF TO data,  
      gs\_data        TYPE REF TO data,  
  
\*&---------------------------------------------------------------------\*  
\*&Declarations for Result Count                                        \*  
\*&---------------------------------------------------------------------\*  
      gv\_success     TYPE zsuccess,  
      gv\_error       TYPE zsuccess,  
  
\*&---------------------------------------------------------------------\*  
\*&Declarations for Factory ALV                                         \*  
\*&---------------------------------------------------------------------\*  
      i\_container    TYPE REF TO cl\_gui\_custom\_container,   " Custom container for ALV holding  
      i\_alvgrid      TYPE REF TO cl\_gui\_alv\_grid,   " class for OOALV grid display  
  
      lv\_STRING      TYPE string,  
      lt\_fieldcat    TYPE lvc\_t\_fcat,  
      la\_fieldcat    TYPE lvc\_s\_fcat,  
      lo\_table       TYPE REF TO data,  
  
\*----------------------------------------------------------------------\*  
      lv\_string2     TYPE string,  
      lv\_string3     TYPE string,  
      lv\_string5     TYPE string,  
  
      IT\_dATASTR     TYPE ztest\_tt,  
      wa\_datstr      TYPE ztest1111,  
  
      IT\_dATASTR1    TYPE ztest\_tt,  
      wa\_datstr1     TYPE ztest1111.  
  
FIELD-SYMBOLS: <ft\_data> TYPE STANDARD TABLE,  
               <fs\_data> TYPE any,  
               <lv\_line> TYPE dfies-fieldname.

\*&---------------------------------------------------------------------\*  
\*& Include          ZGRP4\_SELECTIONSCR\_IP                              \*  
\*&---------------------------------------------------------------------\*  
  
  
\*&---------------------------------------------------------------------\*  
\*& Selection Screen Design and Processing                                          \*  
\*&---------------------------------------------------------------------\*  
SELECTION-SCREEN BEGIN  OF BLOCK  b1 WITH FRAME .  
  PARAMETERS: p\_table TYPE ddobjname OBLIGATORY.  
SELECTION-SCREEN END OF BLOCK b1.  
  
SELECTION-SCREEN SKIP.  
  
SELECTION-SCREEN begin of block  b2 WITH FRAME.  
  PARAMETERS: p\_file LIKE rlgrap-filename .  
SELECTION-SCREEN END OF BLOCK b2.  
  
  
SELECTION-SCREEN SKIP.  
  
SELECTION-SCREEN BEGIN OF BLOCK b3 WITH FRAME.  
  SELECTION-SCREEN: BEGIN OF LINE.  
    SELECTION-SCREEN : COMMENT 1(20) comm1.  
    SELECTION-SCREEN: PUSHBUTTON 35(40) TEXT-002 USER-COMMAND TEMPLATE.  
  SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN END OF BLOCK b3.  
  
INITIALIZATION.  
COMM1 = text-016.  "'File Template'.  
  
  
AT SELECTION-SCREEN ON p\_table.  
 SELECT SINGLE TABNAME FROM dd02l  
  INTO CORRESPONDING FIELDS OF dd02l  
  WHERE tabname = p\_table.  
  CHECK syst-subrc NE 0.  
  MESSAGE e402(mo) WITH p\_table.  
  
  
AT SELECTION-SCREEN ON VALUE-REQUEST FOR p\_file.  
  PERFORM f4\_filename.  
  
\*&--------------------------------------------------------------------------\*  
\*& Form f4\_filename - Process User to select any XL file from local machine \*  
\*&--------------------------------------------------------------------------\*  
FORM f4\_filename.  
  
  CALL FUNCTION 'KD\_GET\_FILENAME\_ON\_F4'  
    EXPORTING  
      program\_name  = syst-repid  
      dynpro\_number = syst-dynnr  
    CHANGING  
      file\_name     = p\_file  
    EXCEPTIONS  
      mask\_too\_long = 1  
      OTHERS        = 2.  
  
  IF sy-subrc <> 0.  
  
    MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
        WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
  
  ENDIF.  
  
ENDFORM.

\*&---------------------------------------------------------------------\*  
\*& Include          ZGRP4\_INPUT\_TEMPLATE  
\*&---------------------------------------------------------------------\*  
  
\*&-----------Process at User Interaction - User Command ---------------\*  
AT SELECTION-SCREEN .  
  IF sy-ucomm EQ 'TEMPLATE'.  
    PERFORM downloadtemplate.  
  ENDIF.  
  
\*&---------------------------------------------------------------------\*  
\*& Form downloadtemplate  
\*&---------------------------------------------------------------------\*  
FORM downloadtemplate.  
  lv\_tabname = p\_table.  
  CONCATENATE lv\_path lv\_tabname lv\_xtnsn INTO lv\_fullpath.  
  
  TRY.  
      CALL FUNCTION 'DDIF\_FIELDINFO\_GET'      "Table field names into int table  
        EXPORTING  
          tabname        = p\_table  
        TABLES  
          dfies\_tab      = lt\_fields  
        EXCEPTIONS  
          not\_found      = 0  
          internal\_error = 0  
          OTHERS         = 0.  
  
      LOOP AT lt\_fields INTO DATA(ls\_field).  "Selected/targetted attributes of the table columns  
                                                 "eg: col-name, col-length, col-datatype, language  
        MOVE: ls\_field-fieldname TO wa\_headings-fields,  
              ls\_field-leng TO wa\_headings-size,  
              ls\_field-datatype TO wa\_headings-dtype.  
        APPEND wa\_headings TO lt\_headings.  
  
      ENDLOOP.  
  
      DATA: lo\_excel      TYPE REF TO zcl\_excel,         "type ref to generate local excel sheet  
            lo\_excel\_conv TYPE REF TO zcl\_excel\_converter.  "i-tab data to excel ready fromat  
  
  
      lo\_excel\_conv = NEW zcl\_excel\_converter( ).  
      lo\_excel\_conv->convert( EXPORTING  
                              it\_table = lt\_headings  
                            CHANGING  
                              co\_excel = lo\_excel ).  
  
      DATA(lo\_worksheet) = lo\_excel->get\_active\_worksheet( ).  "generate excel worksheet.  
      DATA(lo\_excel\_writer) = CAST zif\_excel\_writer( NEW zcl\_excel\_writer\_2007( ) ).  
                                  "writing raw data into excel data- FORMATTING  
  
      DATA(lv\_excel\_data) = lo\_excel\_writer->write\_file( lo\_excel ).  
      DATA(lt\_raw\_data) = cl\_bcs\_convert=>xstring\_to\_solix( EXPORTING  
                                                              iv\_xstring = lv\_excel\_data ).  
  
      cl\_gui\_frontend\_services=>gui\_download( EXPORTING  
                                              filename = lv\_fullpath  
                                              filetype = 'BIN'  
                                              bin\_filesize = xstrlen( lv\_excel\_data )  
                                            CHANGING  
                                              data\_tab = lt\_raw\_data ).  
                                         "Download Ready Excel File into the local machine  
                                                      "with specific naming convention  
      CLEAR lt\_raw\_data.  
      CLEAR lt\_headings.        "Clear memory or int tables to avoid data junk/over-writings/data corruption  
  
    CATCH cx\_root INTO DATA(ex\_txt).      "Catch unwanted Exceptions in case of Performance failure  
      WRITE:/ ex\_txt->get\_text( ).  
  ENDTRY.  
  
ENDFORM.

START-OF-SELECTION.  
\*--------------------- Create dynamic internal table ------------------\*  
  CREATE DATA gt\_data TYPE TABLE OF (p\_table).  
  ASSIGN gt\_data->\* TO <ft\_data>.  
  DATA lt\_raw\_data    TYPE truxs\_t\_text\_data.  
  
  
\*------------------ Create dynamic work area ------------------\*  
  CREATE DATA gs\_data TYPE (p\_table).  
  ASSIGN gs\_data->\* TO <fs\_data>.  
  
  
\*------------------ Get Excel data in internal table ----------\*  
  CALL FUNCTION 'TEXT\_CONVERT\_XLS\_TO\_SAP'  
    EXPORTING  
      i\_line\_header        = 'X'  
      i\_tab\_raw\_data       = lt\_raw\_data  
      i\_filename           = P\_file  
    TABLES  
      i\_tab\_converted\_data = <ft\_data>  
    EXCEPTIONS  
      conversion\_failed    = 1  
      OTHERS               = 2.  
  
  IF sy-subrc <> 0.  
                                  "Handling no-data scenario..  
    MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
                WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
  
  ENDIF.

\*&---------------------------------------------------------------------\*  
\*& Include          ZGRP4\_UPLOAD\_TO\_DB  
\*&---------------------------------------------------------------------\*  
  
  
  
\*---------------------Upload data to database table--------------------\*  
\*&---------------------------------------------------------------------\*  
\*&      Form  upload\_data - ITAB TO DATABASE(MODIFY)  
\*&---------------------------------------------------------------------\*  
  
SET UPDATE TASK LOCAL.   " Switch to local update  
lv\_string3 = sy-uname.  
  
  LOOP AT <ft\_data> ASSIGNING <fs\_data>.  
  
  
    varkey = sy-mandt.  "variable to lock tables based on client  
    CALL FUNCTION 'ENQUEUE\_E\_TABLE'   "Enqueue or LOCK table before modifying  
     EXPORTING  
       TABNAME              = p\_table  
       VARKEY               = varkey  
     EXCEPTIONS  
       FOREIGN\_LOCK         = 0  
       SYSTEM\_FAILURE       = 0  
       OTHERS               = 0 .  
\*   &------------"Modify DATABASE TABLE----------------&  
    MODIFY (p\_table) FROM <fs\_data>.  
  
    CALL FUNCTION 'DEQUEUE\_E\_TABLE'     "Dequeue or UNLOCK Databse Table after updating row entry  
     EXPORTING  
       TABNAME            = p\_table  
       VARKEY             = varkey .  
  
\*    IF sy-subrc = 0.  
      SELECT TABNAME , FIELDNAME  FROM dd03l INTO TABLE @DATA(it\_tab) WHERE tabname = @p\_table AND keyflag = 'X'.  
      LOOP AT it\_tab INTO DATA(wa\_tab11).  
        IF wa\_tab11-fieldname = text-009.  
          CONTINUE.  
        ELSE.  
          ASSIGN COMPONENT wa\_tab11-fieldname OF STRUCTURE <fs\_data> TO  FIELD-SYMBOL(<fk\_data>).  
          IF lv\_string IS INITIAL.  
            lv\_string = <fk\_data>.  
          ELSE.  
            CONCATENATE lv\_string <fk\_data> INTO lv\_string SEPARATED BY space.  
          ENDIF.  
        ENDIF.  
      ENDLOOP.  
  
      wa\_datstr1-col1 = lv\_string.  
      CLEAR lv\_string2.  
      CLEAR lv\_string.  
      APPEND wa\_datstr1 TO IT\_dATASTR1.  
  
      gv\_success = gv\_success + 1.  
  
      CONCATENATE sy-uname text-008 INTO lv\_string5 SEPARATED BY space.  
      wa\_datstr-col1 = lv\_string5.  
      APPEND wa\_datstr TO IT\_dATASTR.  
  
  ENDLOOP.

&---------------------------------------------------------------------\*  
\*& Include          ZGRP4\_DISPLAY\_STATUS  
\*&---------------------------------------------------------------------\*  
  
\*------------------Create Factory ALV and display result---------------\*  
END-OF-SELECTION.  
CALL SCREEN 100.

PROCESS BEFORE OUTPUT.  
 MODULE STATUS\_0100.  
 MODULE OOALV\_INFO.  
 MODULE OOALV\_IMAGE.  
\*  
PROCESS AFTER INPUT.  
 MODULE USER\_COMMAND\_0100.

&---------------------------------------------------------------------\*  
\*& Module STATUS\_0100 OUTPUT  
\*&---------------------------------------------------------------------\*  
MODULE status\_0100 OUTPUT.  
 SET PF-STATUS SPACE.  
ENDMODULE.  
  
\*&---------------------------------------------------------------------\*  
\*& Module OOALV\_INFO OUTPUT  
\*&---------------------------------------------------------------------\*  
MODULE ooalv\_info OUTPUT.  
  
create object i\_container  
    exporting  
      container\_name = 'CTN1'  " SCREEN NAME  
  exceptions  
     cntl\_error     = 1  
     cntl\_system\_error           = 2  
     create\_error   = 3  
     lifetime\_error = 4  
     lifetime\_dynpro\_dynpro\_link = 5  
     others         = 6  
    .  
  if sy-subrc <> 0.  
         MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO  
                WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.  
  endif.  
  
  create object i\_alvgrid  
    exporting  
      i\_parent = i\_container  
  exceptions  
     error\_cntl\_create = 1  
     error\_cntl\_init   = 2  
     error\_cntl\_link   = 3  
     error\_dp\_create   = 4  
     others   = 5  
          .  
  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 i\_alvgrid->set\_table\_for\_first\_display  
  EXPORTING  
    i\_structure\_name              = p\_table  
  CHANGING  
    it\_outtab                     = <FT\_DATA>  
  EXCEPTIONS  
    invalid\_parameter\_combination = 1  
    program\_error                 = 2  
    too\_many\_lines                = 3  
    others                        = 4  
        .  
IF sy-subrc <> 0.  
\* Implement suitable error handling here  
ENDIF.  
  
  
ENDMODULE.  
\*&---------------------------------------------------------------------\*  
\*& Module OOALV\_IMAGE OUTPUT  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
MODULE ooalv\_image OUTPUT.  
  
  DATA: W\_LINES TYPE I.  
  TYPES PICT\_LINE(256) TYPE C.  
  DATA :  
        CONTAINER TYPE REF TO CL\_GUI\_CUSTOM\_CONTAINER,  
        EDITOR TYPE REF TO CL\_GUI\_TEXTEDIT,  
        PICTURE TYPE REF TO CL\_GUI\_PICTURE,  
        PICT\_TAB TYPE TABLE OF PICT\_LINE,  
        URL(255) TYPE C.  
  DATA: GRAPHIC\_URL(255).  
  DATA: BEGIN OF GRAPHIC\_TABLE OCCURS 0,  
    LINE(255) TYPE X,  
  END OF GRAPHIC\_TABLE.  
  DATA: L\_GRAPHIC\_CONV TYPE I.  
  DATA: L\_GRAPHIC\_OFFS TYPE I.  
  DATA: GRAPHIC\_SIZE TYPE I.  
  DATA: L\_GRAPHIC\_XSTR TYPE XSTRING.  
  .  
  CALL METHOD CL\_GUI\_CFW=>FLUSH.  
  CREATE OBJECT:  
  
\*  CONTAINER EXPORTING CONTAINER\_NAME = 'PICTURE\_CONTAINER',  
  CONTAINER EXPORTING CONTAINER\_NAME = 'CTN2',       "  # 1 call screeen CONT NAME  
  
  PICTURE EXPORTING PARENT = CONTAINER.  
  CALL METHOD CL\_SSF\_XSF\_UTILITIES=>GET\_BDS\_GRAPHIC\_AS\_BMP  
  EXPORTING  
    P\_OBJECT       = 'GRAPHICS'  
    P\_NAME         = 'Z353\_UST\_LOGO'        " # 2 SE78 T-CODE NAME  
    P\_ID           = 'BMAP'  
    P\_BTYPE        = 'BCOL'  
    RECEIVING  
    P\_BMP          = L\_GRAPHIC\_XSTR  
  EXCEPTIONS  
    NOT\_FOUND      = 1  
    INTERNAL\_ERROR = 2  
    others         = 3  
    .  
  IF SY-SUBRC <> 0.  
\* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO  
\*            WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.  
  ENDIF.  
  
  GRAPHIC\_SIZE = XSTRLEN( L\_GRAPHIC\_XSTR ).  
  L\_GRAPHIC\_CONV = GRAPHIC\_SIZE.  
  L\_GRAPHIC\_OFFS = 0.  
  WHILE L\_GRAPHIC\_CONV > 255.  
    GRAPHIC\_TABLE-LINE = L\_GRAPHIC\_XSTR+L\_GRAPHIC\_OFFS(255).  
    APPEND GRAPHIC\_TABLE.  
    L\_GRAPHIC\_OFFS = L\_GRAPHIC\_OFFS + 255.  
    L\_GRAPHIC\_CONV = L\_GRAPHIC\_CONV - 255.  
  ENDWHILE.  
  GRAPHIC\_TABLE-LINE = L\_GRAPHIC\_XSTR+L\_GRAPHIC\_OFFS(L\_GRAPHIC\_CONV).  
  APPEND GRAPHIC\_TABLE.  
  CALL FUNCTION 'DP\_CREATE\_URL'  
  EXPORTING  
    TYPE     = 'IMAGE'  
    SUBTYPE  = 'X-UNKNOWN'  
    SIZE     = GRAPHIC\_SIZE  
    LIFETIME = 'T'  
  TABLES  
  DATA     = GRAPHIC\_TABLE  
  CHANGING  
    URL      = URL.  
  CALL METHOD PICTURE->LOAD\_PICTURE\_FROM\_URL  
  EXPORTING  
    URL = URL.  
  CALL METHOD PICTURE->SET\_DISPLAY\_MODE  
  EXPORTING  
    DISPLAY\_MODE = PICTURE->DISPLAY\_MODE\_FIT\_CENTER.  
  
  
ENDMODULE.  
  
\*&---------------------------------------------------------------------\*  
\*&      Module  USER\_COMMAND\_0100  INPUT  
\*&---------------------------------------------------------------------\*  
MODULE user\_command\_0100 INPUT.  
  data ok\_code type sy-ucomm.  
  
  CASE ok\_code.  
    when 'BACK'.  
      leave to screen 0.  
    WHEN 'FORM'.  
      PERFORM DISPLAY\_FORM.  
    WHEN 'DISPLAY'.  
      CALL TRANSACTION 'ZG04\_CDS'.  
   ENDCASE.  
  
ENDMODULE.  
\*&---------------------------------------------------------------------\*  
\*& Form DISPLAY\_FORM  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM display\_form .  
  DATA : WA\_OUTPARAMS TYPE SFPOUTPUTPARAMS,  
         FMNAME TYPE FPNAME.  
  
  wA\_OUTPARAMS-deVICE = 'PRINTER'.  
  wA\_OUTPARAMS-dest = 'LP01'.  
  wA\_OUTPARAMS-nodialog = 'X'.  
  wA\_OUTPARAMS-preview = 'X'.  
  
  
\*  CALL FUNCTION 'FP\_JOB\_OPEN'  
\*    CHANGING  
\*      ie\_outputparams       = WA\_OUTPARAMS  
\*    .  
CALL FUNCTION 'FP\_JOB\_OPEN'  
  CHANGING  
    ie\_outputparams       = wa\_outparams  
 EXCEPTIONS  
   CANCEL                = 0  
   USAGE\_ERROR           = 0  
   SYSTEM\_ERROR          = 0  
   INTERNAL\_ERROR        = 0  
   OTHERS                = 0  
          .  
try.  
  
  CALL FUNCTION 'FP\_FUNCTION\_MODULE\_NAME'  
    EXPORTING  
      i\_name                     = 'ZGRP4\_TEST2'  
   IMPORTING  
     E\_FUNCNAME                 = fmname  
            .  
  CATCH CX\_ROOT INTO DATA(EX\_TXT).  
    WRITE:/ EX\_TXT->get\_text( ).  
  endtry.  
  
  
  
  
  
  CALL FUNCTION fmname "'/1BCDWB/SM00001130'  
    EXPORTING  
      it\_tab1                  = IT\_dATASTR  
      it\_tab                   = IT\_dATASTR1  
      p\_table                  = p\_table  
            .  
  
\* IMPORTING  
\*   /1BCDWB/FORMOUTPUT       =  
\* EXCEPTIONS  
\*   USAGE\_ERROR              = 1  
\*   SYSTEM\_ERROR             = 2  
\*   INTERNAL\_ERROR           = 3  
\*   OTHERS                   = 4  
          .  
\*IF sy-subrc <> 0.  
\*\* Implement suitable error handling here  
\*ENDIF.  
  
  
  
  CALL FUNCTION 'FP\_JOB\_CLOSE'  
\*   EXCEPTIONS  
\*     USAGE\_ERROR          = 1  
\*     SYSTEM\_ERROR         = 2  
\*     INTERNAL\_ERROR       = 3  
\*     OTHERS               = 4  
\*            .  
.  
  
ENDFORM.

\*----------------------------------------------------------------------\*  
\*\*\*INCLUDE ZGRP4\_EXCEL\_ALV\_STATUS\_0100O01.  
\*----------------------------------------------------------------------\*  
  
  
\*&---------------------------------------------------------------------\*  
\*& Module STATUS\_0100 OUTPUT  
\*&---------------------------------------------------------------------\*  
MODULE status\_0100 OUTPUT.  
 SET PF-STATUS SPACE.  
ENDMODULE.  
  
\*&---------------------------------------------------------------------\*  
\*& Module OOALV\_INFO OUTPUT  
\*&---------------------------------------------------------------------\*  
MODULE ooalv\_info OUTPUT.  
  
create object i\_container  
    exporting  
      container\_name = 'CTN1'  " SCREEN NAME  
  exceptions  
     cntl\_error     = 1  
     cntl\_system\_error           = 2  
     create\_error   = 3  
     lifetime\_error = 4  
     lifetime\_dynpro\_dynpro\_link = 5  
     others         = 6  
    .  
  if sy-subrc <> 0.  
         MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO  
                WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.  
  endif.  
  
  create object i\_alvgrid  
    exporting  
      i\_parent = i\_container  
  exceptions  
     error\_cntl\_create = 1  
     error\_cntl\_init   = 2  
     error\_cntl\_link   = 3  
     error\_dp\_create   = 4  
     others   = 5  
          .  
  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 i\_alvgrid->set\_table\_for\_first\_display  
  EXPORTING  
    i\_structure\_name              = p\_table  
  CHANGING  
    it\_outtab                     = <FT\_DATA>  
  EXCEPTIONS  
    invalid\_parameter\_combination = 1  
    program\_error                 = 2  
    too\_many\_lines                = 3  
    others                        = 4  
        .  
IF sy-subrc <> 0.  
\* Implement suitable error handling here  
ENDIF.  
  
  
ENDMODULE.  
\*&---------------------------------------------------------------------\*  
\*& Module OOALV\_IMAGE OUTPUT  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
MODULE ooalv\_image OUTPUT.  
  
  DATA: W\_LINES TYPE I.  
  TYPES PICT\_LINE(256) TYPE C.  
  DATA :  
        CONTAINER TYPE REF TO CL\_GUI\_CUSTOM\_CONTAINER,  
        EDITOR TYPE REF TO CL\_GUI\_TEXTEDIT,  
        PICTURE TYPE REF TO CL\_GUI\_PICTURE,  
        PICT\_TAB TYPE TABLE OF PICT\_LINE,  
        URL(255) TYPE C.  
  DATA: GRAPHIC\_URL(255).  
  DATA: BEGIN OF GRAPHIC\_TABLE OCCURS 0,  
    LINE(255) TYPE X,  
  END OF GRAPHIC\_TABLE.  
  DATA: L\_GRAPHIC\_CONV TYPE I.  
  DATA: L\_GRAPHIC\_OFFS TYPE I.  
  DATA: GRAPHIC\_SIZE TYPE I.  
  DATA: L\_GRAPHIC\_XSTR TYPE XSTRING.  
  .  
  CALL METHOD CL\_GUI\_CFW=>FLUSH.  
  CREATE OBJECT:  
  
\*  CONTAINER EXPORTING CONTAINER\_NAME = 'PICTURE\_CONTAINER',  
  CONTAINER EXPORTING CONTAINER\_NAME = 'CTN2',       "  # 1 call screeen CONT NAME  
  
  PICTURE EXPORTING PARENT = CONTAINER.  
  CALL METHOD CL\_SSF\_XSF\_UTILITIES=>GET\_BDS\_GRAPHIC\_AS\_BMP  
  EXPORTING  
    P\_OBJECT       = 'GRAPHICS'  
    P\_NAME         = 'Z353\_UST\_LOGO'        " # 2 SE78 T-CODE NAME  
    P\_ID           = 'BMAP'  
    P\_BTYPE        = 'BCOL'  
    RECEIVING  
    P\_BMP          = L\_GRAPHIC\_XSTR  
  EXCEPTIONS  
    NOT\_FOUND      = 1  
    INTERNAL\_ERROR = 2  
    others         = 3  
    .  
  IF SY-SUBRC <> 0.  
\* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO  
\*            WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.  
  ENDIF.  
  
  GRAPHIC\_SIZE = XSTRLEN( L\_GRAPHIC\_XSTR ).  
  L\_GRAPHIC\_CONV = GRAPHIC\_SIZE.  
  L\_GRAPHIC\_OFFS = 0.  
  WHILE L\_GRAPHIC\_CONV > 255.  
    GRAPHIC\_TABLE-LINE = L\_GRAPHIC\_XSTR+L\_GRAPHIC\_OFFS(255).  
    APPEND GRAPHIC\_TABLE.  
    L\_GRAPHIC\_OFFS = L\_GRAPHIC\_OFFS + 255.  
    L\_GRAPHIC\_CONV = L\_GRAPHIC\_CONV - 255.  
  ENDWHILE.  
  GRAPHIC\_TABLE-LINE = L\_GRAPHIC\_XSTR+L\_GRAPHIC\_OFFS(L\_GRAPHIC\_CONV).  
  APPEND GRAPHIC\_TABLE.  
  CALL FUNCTION 'DP\_CREATE\_URL'  
  EXPORTING  
    TYPE     = 'IMAGE'  
    SUBTYPE  = 'X-UNKNOWN'  
    SIZE     = GRAPHIC\_SIZE  
    LIFETIME = 'T'  
  TABLES  
  DATA     = GRAPHIC\_TABLE  
  CHANGING  
    URL      = URL.  
  CALL METHOD PICTURE->LOAD\_PICTURE\_FROM\_URL  
  EXPORTING  
    URL = URL.  
  CALL METHOD PICTURE->SET\_DISPLAY\_MODE  
  EXPORTING  
    DISPLAY\_MODE = PICTURE->DISPLAY\_MODE\_FIT\_CENTER.  
  
  
ENDMODULE.  
  
\*&---------------------------------------------------------------------\*  
\*&      Module  USER\_COMMAND\_0100  INPUT  
\*&---------------------------------------------------------------------\*  
MODULE user\_command\_0100 INPUT.  
  data ok\_code type sy-ucomm.  
  
  CASE ok\_code.  
    when 'BACK'.  
      leave to screen 0.  
    WHEN 'FORM'.  
      PERFORM DISPLAY\_FORM.  
    WHEN 'DISPLAY'.  
      CALL TRANSACTION 'ZG04\_CDS'.  
   ENDCASE.  
  
ENDMODULE.  
\*&---------------------------------------------------------------------\*  
\*& Form DISPLAY\_FORM  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM display\_form .  
  DATA : WA\_OUTPARAMS TYPE SFPOUTPUTPARAMS,  
         FMNAME TYPE FPNAME.  
  
  wA\_OUTPARAMS-deVICE = 'PRINTER'.  
  wA\_OUTPARAMS-dest = 'LP01'.  
  wA\_OUTPARAMS-nodialog = 'X'.  
  wA\_OUTPARAMS-preview = 'X'.  
  
  
\*  CALL FUNCTION 'FP\_JOB\_OPEN'  
\*    CHANGING  
\*      ie\_outputparams       = WA\_OUTPARAMS  
\*    .  
CALL FUNCTION 'FP\_JOB\_OPEN'  
  CHANGING  
    ie\_outputparams       = wa\_outparams  
 EXCEPTIONS  
   CANCEL                = 0  
   USAGE\_ERROR           = 0  
   SYSTEM\_ERROR          = 0  
   INTERNAL\_ERROR        = 0  
   OTHERS                = 0  
          .  
try.  
  
  CALL FUNCTION 'FP\_FUNCTION\_MODULE\_NAME'  
    EXPORTING  
      i\_name                     = 'ZGRP4\_TEST2'  
   IMPORTING  
     E\_FUNCNAME                 = fmname  
            .  
  CATCH CX\_ROOT INTO DATA(EX\_TXT).  
    WRITE:/ EX\_TXT->get\_text( ).  
  endtry.  
  
  
  
  
  
  CALL FUNCTION fmname "'/1BCDWB/SM00001130'  
    EXPORTING  
      it\_tab1                  = IT\_dATASTR  
      it\_tab                   = IT\_dATASTR1  
      p\_table                  = p\_table  
            .  
  
\* IMPORTING  
\*   /1BCDWB/FORMOUTPUT       =  
\* EXCEPTIONS  
\*   USAGE\_ERROR              = 1  
\*   SYSTEM\_ERROR             = 2  
\*   INTERNAL\_ERROR           = 3  
\*   OTHERS                   = 4  
          .  
\*IF sy-subrc <> 0.  
\*\* Implement suitable error handling here  
\*ENDIF.  
  
  
  
  CALL FUNCTION 'FP\_JOB\_CLOSE'  
\*   EXCEPTIONS  
\*     USAGE\_ERROR          = 1  
\*     SYSTEM\_ERROR         = 2  
\*     INTERNAL\_ERROR       = 3  
\*     OTHERS               = 4  
\*            .  
.  
  
ENDFORM.

\*\*\*\*\*\*\*\*\*&&&&&&&& **CDS VIEW CODE SNIPPET**&&&&&&&\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

@AbapCatalog.sqlViewName: 'ZGO4\_DDIC\_CDS'

@AbapCatalog.compiler.compareFilter: true

@AbapCatalog.preserveKey: true

@AccessControl.authorizationCheck: #NOT\_REQUIRED

@EndUserText.label: 'CDS view on Employee table-Group 4'

**define** **view** ZG04\_CDS

**with** **parameters** P\_EMP1 **:** *zgo4\_empno***,**

P\_EMP2 **:** *zgo4\_empno*

**as** **select** **from** zg04\_employee **as** EMP\_TAB

**{**

**key** emp\_no **as** ID**,**

emp\_first\_name **as** FNAME**,**

emp\_last\_name **as** LNAME

**}** **where** EMP\_TAB**.**emp\_no **between** **:**P\_EMP1 **and** **:**P\_EMP2**;**