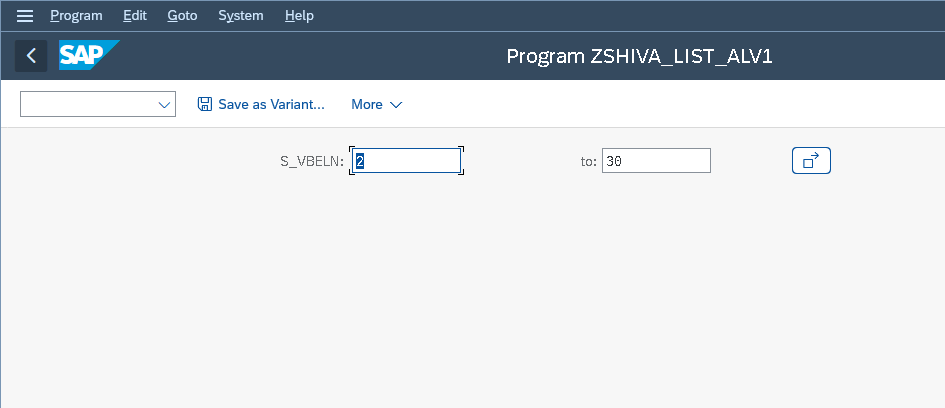
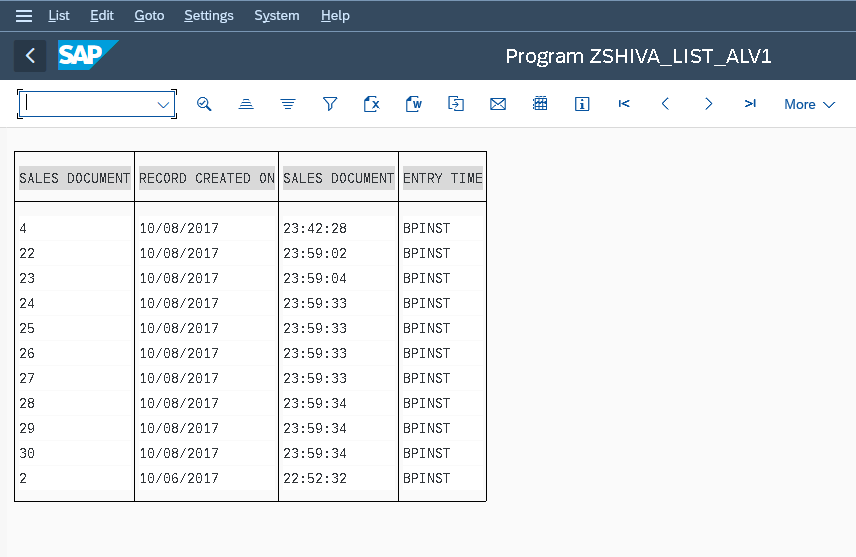
WEEK 3 PROGRAMS:

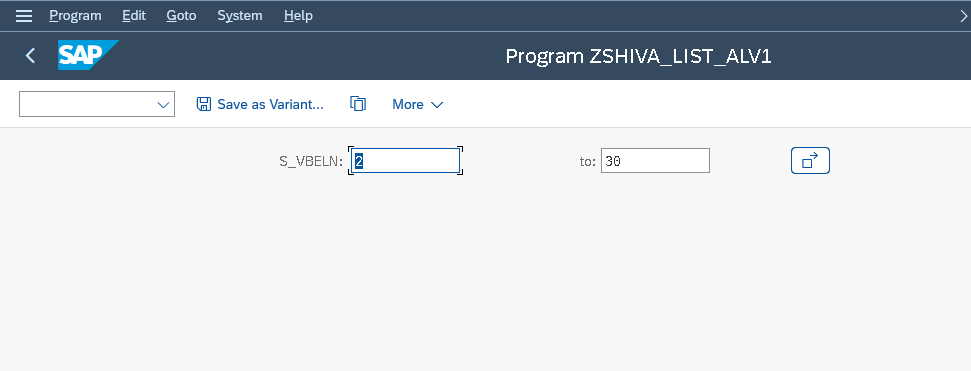
\*&---------------------------------------------------------------------\*  
\*& Report ZSHIVA\_LIST\_ALV1  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT zshiva\_list\_alv1 NO STANDARD PAGE HEADING LINE-COUNT 22(3) LINE-SIZE 350.  
  
INCLUDE  zshiva\_vbak\_inc. " include program with structure of table  
  
  
SELECT-OPTIONS s\_vbeln FOR lv\_vbeln. "select range for prmary key  
  
INITIALIZATION. " initialization for range start and range end  
  CLEAR : lt\_tab1, wa\_fieldcat, lt\_fieldcat.  
  s\_vbeln-low = '2'.  
  s\_vbeln-high = '30'.  
  APPEND s\_vbeln.  " append the low high to S\_VBELN.  
  
AT SELECTION-SCREEN ON s\_vbeln . "FIRST OUTPUT SCREEN  WITH RANGE  
  
  SELECT SINGLE vbeln  " RANGE VALIDATION INTO LOCAL DATA LV\_VBELN ,FROM TABLE  
    INTO lv\_vbeln  
    FROM vbak  
  WHERE vbeln IN s\_vbeln.  
  
  IF sy-subrc EQ 0.  
    MESSAGE S000(zshiva\_eror\_msg).  
  ELSE.  
    MESSAGE E001(zhiva\_eror\_msg).  
  ENDIF.  
  
START-OF-SELECTION.  
  "PERFORM get\_alav\_fieldcat.  
  
  PERFORM get\_vbeln\_data.  
  PERFORM get\_alav\_fieldcat.  
  
END-OF-SELECTION.  
  
  PERFORM display\_vbeln\_alvout.  
  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_DATA  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALAV\_FIELDCAT  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM get\_alav\_fieldcat .  
  """"""VBELN.  
  wa\_fieldcat-col\_pos = '1'.  
  wa\_fieldcat-seltext\_l = 'SALES DOCUMENT'.  
  wa\_fieldcat-fieldname = 'VBELN'.  
  wa\_fieldcat-tabname = 'LT\_TAB1'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  """"""ERDAT.  
  wa\_fieldcat-col\_pos = '2'.  
  wa\_fieldcat-seltext\_l = 'RECORD CREATED ON'.  
  wa\_fieldcat-fieldname = 'ERDAT'.  
  wa\_fieldcat-tabname = 'LT\_TAB1'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  """"""ERZET.  
  wa\_fieldcat-col\_pos = '3'.  
  wa\_fieldcat-seltext\_l = 'SALES DOCUMENT'.  
  wa\_fieldcat-fieldname = 'ERZET'.  
  wa\_fieldcat-tabname = 'LT\_TAB1'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  """"""ERNUM.  
  wa\_fieldcat-col\_pos = '4'.  
  wa\_fieldcat-seltext\_l = 'ENTRY TIME '.  
  wa\_fieldcat-fieldname = 'ERNAM'.  
  wa\_fieldcat-tabname = 'LT\_TAB1'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_VBELN\_DATA  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM get\_vbeln\_data .  
  SELECT vbeln erdat erzet ernam  
    INTO TABLE lt\_tab1  
  FROM vbak  
  WHERE vbeln IN s\_vbeln.  
  IF sy-subrc EQ 0.  
    MESSAGE s000(zshiva\_eror\_msg).  
  ELSE.  
    MESSAGE s001(zshiva\_eror\_msg).  
  ENDIF.  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form DISPLAY\_VBELN\_ALVOUT  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM display\_vbeln\_alvout .  
  CALL FUNCTION 'REUSE\_ALV\_LIST\_DISPLAY'  
    EXPORTING  
\*     I\_INTERFACE\_CHECK  = ' '  
\*     I\_BYPASSING\_BUFFER =  
\*     I\_BUFFER\_ACTIVE    = ' '  
      i\_callback\_program = SY-REPID  
\*     I\_CALLBACK\_PF\_STATUS\_SET       = ' '  
\*     I\_CALLBACK\_USER\_COMMAND        = ' '  
\*     I\_STRUCTURE\_NAME   =  
\*     IS\_LAYOUT          =  
      it\_fieldcat        = lt\_fieldcat  
\*     IT\_EXCLUDING       =  
\*     IT\_SPECIAL\_GROUPS  =  
\*     IT\_SORT            =  
\*     IT\_FILTER          =  
\*     IS\_SEL\_HIDE        =  
\*     I\_DEFAULT          = 'X'  
\*     I\_SAVE             = ' '  
\*     IS\_VARIANT         =  
\*     IT\_EVENTS          =  
\*     IT\_EVENT\_EXIT      =  
\*     IS\_PRINT           =  
\*     IS\_REPREP\_ID       =  
\*     I\_SCREEN\_START\_COLUMN          = 0  
\*     I\_SCREEN\_START\_LINE            = 0  
\*     I\_SCREEN\_END\_COLUMN            = 0  
\*     I\_SCREEN\_END\_LINE  = 0  
\*     IR\_SALV\_LIST\_ADAPTER           =  
\*     IT\_EXCEPT\_QINFO    =  
\*     I\_SUPPRESS\_EMPTY\_DATA          = ABAP\_FALSE  
\*     IO\_SALV\_ADAPTER    =  
\*     O\_COMMON\_HUB       =  
\* IMPORTING  
\*     E\_EXIT\_CAUSED\_BY\_CALLER        =  
\*     ES\_EXIT\_CAUSED\_BY\_USER         =  
    TABLES  
      t\_outtab           = lt\_tab1  
    EXCEPTIONS  
      program\_error      = 1  
      OTHERS             = 2.  
  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
  ENDIF.  
  
  
ENDFORM.



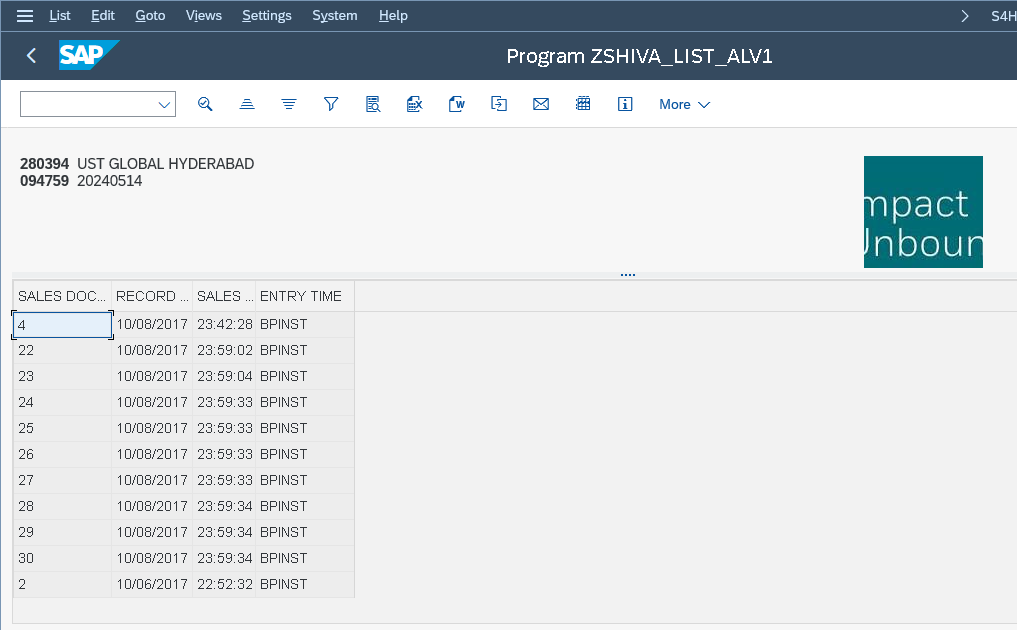
FINAL OUTPUT;



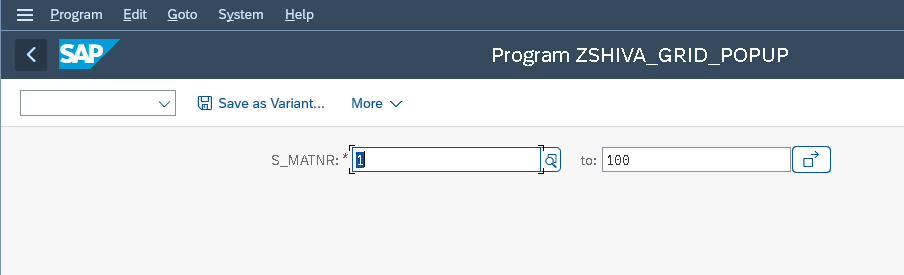
REPORT zshiva\_grid\_alv1 NO STANDARD PAGE HEADING LINE-COUNT 22(3) LINE-SIZE 350.  
INCLUDE ZSHIVA\_VBAK\_GRID\_INC.  
\*INCLUDE  zshiva\_vbak\_inc. " include program with structure of table  
SELECT-OPTIONS s\_vbeln FOR lv\_vbeln. "select range for prmary key  
INITIALIZATION. " initialization for range start and range end  
  CLEAR : lt\_tab1, wa\_fieldcat, lt\_fieldcat.  
  s\_vbeln-low = '2'.  
  s\_vbeln-high = '30'.  
  APPEND s\_vbeln.  " append the low high to S\_VBELN.  
AT SELECTION-SCREEN ON s\_vbeln . "FIRST OUTPUT SCREEN  WITH RANGE  
  SELECT SINGLE vbeln  " RANGE VALIDATION INTO LOCAL DATA LV\_VBELN ,FROM TABLE  
    INTO lv\_vbeln  
    FROM vbak  
  WHERE vbeln IN s\_vbeln.  
  IF sy-subrc EQ 0.  
    MESSAGE S000(zshiva\_eror\_msg).  
  ELSE.  
    MESSAGE E001(zhiva\_eror\_msg).  
  ENDIF.  
START-OF-SELECTION.  
  "PERFORM get\_alav\_fieldcat.  
  PERFORM get\_vbeln\_data.  
  PERFORM get\_alav\_fieldcat.  
END-OF-SELECTION.  
  PERFORM display\_vbeln\_alvout.  
FORM get\_alav\_fieldcat .  
  """"""VBELN.  
  wa\_fieldcat-col\_pos = '1'.  
  wa\_fieldcat-seltext\_l = 'SALES DOCUMENT'.  
  wa\_fieldcat-fieldname = 'VBELN'.  
  wa\_fieldcat-tabname = 'LT\_TAB1'.  
  WA\_FIELDCAT-no\_zero = 'X'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  """"""ERDAT.  
  wa\_fieldcat-col\_pos = '2'.  
  wa\_fieldcat-seltext\_l = 'RECORD CREATED ON'.  
  wa\_fieldcat-fieldname = 'ERDAT'.  
  wa\_fieldcat-tabname = 'LT\_TAB1'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  """"""ERZET.  
  wa\_fieldcat-col\_pos = '3'.  
  wa\_fieldcat-seltext\_l = 'SALES DOCUMENT'.  
  wa\_fieldcat-fieldname = 'ERZET'.  
  wa\_fieldcat-tabname = 'LT\_TAB1'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  """"""ERNUM.  
  wa\_fieldcat-col\_pos = '4'.  
  wa\_fieldcat-seltext\_l = 'ENTRY TIME '.  
  wa\_fieldcat-fieldname = 'ERNAM'.  
  wa\_fieldcat-tabname = 'LT\_TAB1'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
ENDFORM.  
  
FORM get\_vbeln\_data .  
  SELECT vbeln erdat erzet ernam  
    INTO TABLE lt\_tab1  
  FROM vbak  
  WHERE vbeln IN s\_vbeln.  
  IF sy-subrc EQ 0.  
    MESSAGE s000(zshiva\_eror\_msg).  
  ELSE.  
    MESSAGE s001(zshiva\_eror\_msg).  
  ENDIF.  
ENDFORM.  
  
FORM display\_vbeln\_alvout .  
CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
EXPORTING  
  I\_CALLBACK\_PROGRAM                = SY-REPID  
  I\_CALLBACK\_TOP\_OF\_PAGE            = 'UST\_LOGO'  
  IT\_FIELDCAT                       = LT\_FIELDCAT  
  TABLES  
    t\_outtab                          = LT\_TAB1  
EXCEPTIONS  
   PROGRAM\_ERROR                     = 1  
   OTHERS                            = 2  
          .  
IF sy-subrc <> 0.  
\* Implement suitable error handling here  
ENDIF.  
  
  
ENDFORM.  
  
FORM UST\_LOGO.  
  DATA : LT\_HEADER TYPE SLIS\_T\_LISTHEADER,  
        WA\_HEADER TYPE SLIS\_LISTHEADER.  
  
\*  WA\_HEADER-TYP = 'A'.  
\* WA\_HEADER-TYP = 'H'.  
  WA\_HEADER-TYP = 'S'.  
  
  WA\_HEADER-INFO = 'UST GLOBAL HYDERABAD'.  
  WA\_HEADER-KEY = '280394'.  
  APPEND WA\_HEADER TO LT\_HEADER.  
  
  
\*  
  WA\_HEADER-INFO = SY-DATUM.  
  WA\_HEADER-KEY = SY-UZEIT.  
  APPEND WA\_HEADER TO LT\_HEADER.  
  
CALL FUNCTION 'REUSE\_ALV\_COMMENTARY\_WRITE'  
  EXPORTING  
    it\_list\_commentary       = LT\_HEADER  
      I\_LOGO                 = 'ZSHIVA\_LOGO'.  
ENDFORM.



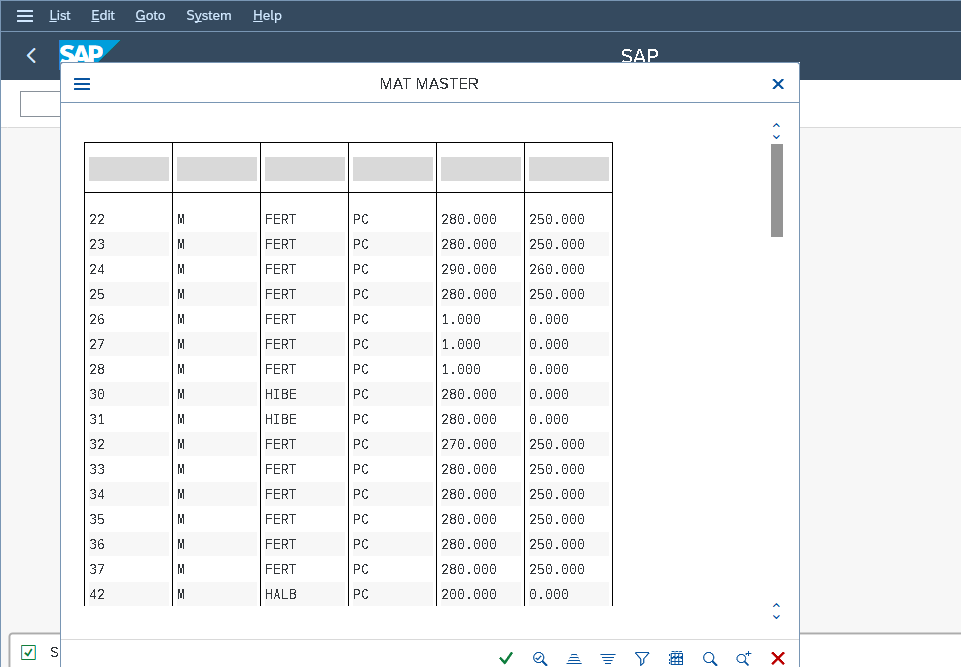
FINAL OUTPUT:



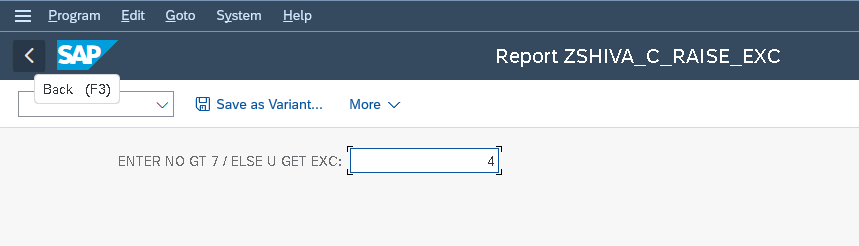
\*&---------------------------------------------------------------------\*  
\*& Report ZSHIVA\_GRID\_POPUP  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSHIVA\_GRID\_POPUP  
"report ztrupen\_ust\_alv\_list\_display  
no standard page heading line-count 20(3)  
line-size 500 message-id zshiva\_eror.  
  
include ztrupen\_ust\_alv\_logo\_top.  
  
select-options s\_matnr for lv\_matnr obligatory.  
  
initialization.  
  clear: lt\_Tab, wa\_fieldcat, lt\_fieldcat.  
  s\_matnr-low  = '1'.  
  s\_matnr-high  = '100'.  
  append s\_matnr.  
  
at selection-screen on s\_matnr.   " 1 to 100  
  
  select single matnr  
    into lv\_matnr  
     from mara  
     where matnr in s\_matnr.  
  
  if sy-subrc eq 0.  
    message s000.  " se91 - tcode.  
  else.  
    message e001.  " se91 - tcode.  
  endif.  
  
start-of-selection.  
  
  perform get\_alv\_fcat.  
  perform get\_mat\_data.  
  
end-of-selection.  
  perform disp\_mat\_alvout.  
  
form get\_mat\_data.  
  select matnr mbrsh mtart meins brgew ntgew  
      into table lt\_Tab  
      from mara  
      where matnr in  s\_matnr.  
  
  if sy-subrc eq 0.  
    message s000.  
  else.  
    message e001.  
  endif.  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALV\_FCAT  
\*&---------------------------------------------------------------------\*  
form get\_alv\_fcat .  
\*\*\*\*\*  MATNR  
\*\*\*\*\*\*\*\*\*\*\*\*  LV\_COUNT = LV\_COUNT + 1.  
  wa\_fieldcat-col\_pos =  '1'.  
  wa\_fieldcat-seltext\_l = text-001. "   'MAT NUMBER'.  
  wa\_fieldcat-fieldname = cs\_matnr. "  'MATNR'.  
  wa\_fieldcat-tabname  = cs\_tab .  "   'LT\_TAB'.  
\*  WA\_FIELDCAT-emphasize = 'C101'.  
  wa\_fieldcat-no\_zero = 'X'.  
  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
\*\*\*\*\*  MBRSH  
  wa\_fieldcat-col\_pos = '2'.  
  wa\_fieldcat-seltext\_l = text-002 ."  'MAT IND SECTOR'.  
  wa\_fieldcat-fieldname = cs\_mbrsh . "  'MBRSH'.  
  wa\_fieldcat-tabname  = cs\_tab .  " 'LT\_TAB'.  
  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
\*\*\*\*\*  MTART  
  wa\_fieldcat-col\_pos = '3'.  
  wa\_fieldcat-seltext\_l = text-003 . "'MAT TYPE'.  
  wa\_fieldcat-fieldname = 'MTART'.  
  wa\_fieldcat-tabname  = cs\_tab."   'LT\_TAB'.  
\*  WA\_FIELDCAT-emphasize = 'C200'.  
  
  
  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  
\*\*\*\*\*  MEINS  
  wa\_fieldcat-col\_pos = '4'.  
  wa\_fieldcat-seltext\_l = text-004 ."'MAT BASE UOM'.  
  wa\_fieldcat-fieldname = 'MEINS'.  
  wa\_fieldcat-tabname  = 'LT\_TAB'.  
\*  WA\_FIELDCAT-emphasize = 'C611'.  
  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  
  
\*\*\*\*\*  BRGEW  
  wa\_fieldcat-col\_pos = '5'.  
  wa\_fieldcat-seltext\_l = text-005. "'MAT GROSS WEIGHT'.  
  wa\_fieldcat-fieldname = 'BRGEW'.  
  wa\_fieldcat-tabname  = cs\_tab . "  'LT\_TAB'.  
\*WA\_FIELDCAT-emphasize = 'C711'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
\*\*\*\*\*  NTGEW  
  wa\_fieldcat-col\_pos = '6'.  
  wa\_fieldcat-seltext\_s = text-006 ."  'MAT NET WEIGHT'.  
  
  wa\_fieldcat-fieldname = 'NTGEW'.  
  wa\_fieldcat-tabname  = cs\_tab ."   'LT\_TAB'.  
\*  WA\_FIELDCAT-emphasize = 'C411'.  
  wa\_fieldcat-do\_sum = 'X'.  
  
  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
endform.  
\*&---------------------------------------------------------------------\*  
\*& Form DISP\_MAT\_ALVOUT  
\*&---------------------------------------------------------------------\*  
\*& text  
form disp\_mat\_alvout .  
  
  
call function 'REUSE\_ALV\_POPUP\_TO\_SELECT'  
  exporting  
   I\_TITLE                       = 'MAT MASTER'  
\*   I\_SELECTION                   = 'X'  
\*   I\_ALLOW\_NO\_SELECTION          =  
   I\_ZEBRA                       = 'X'  
\*   I\_SCREEN\_START\_COLUMN         = 0  
\*   I\_SCREEN\_START\_LINE           = 0  
\*   I\_SCREEN\_END\_COLUMN           = 0  
\*   I\_SCREEN\_END\_LINE             = 0  
\*   I\_CHECKBOX\_FIELDNAME          =  
\*   I\_LINEMARK\_FIELDNAME          =  
\*   I\_SCROLL\_TO\_SEL\_LINE          = 'X'  
    i\_tabname                     =  'LT\_TAB'  
   I\_STRUCTURE\_NAME              = 'LS\_TAB'  
   IT\_FIELDCAT                   = LT\_FIELDCAT  
\*   IT\_EXCLUDING                  =  
   I\_CALLBACK\_PROGRAM            =  SY-REPID  
\*   I\_CALLBACK\_USER\_COMMAND       =  
\*   IS\_PRIVATE                    =  
\* IMPORTING  
\*   ES\_SELFIELD                   =  
\*   E\_EXIT                        =  
  tables  
    t\_outtab                      = LT\_TAB  
\* EXCEPTIONS  
\*   PROGRAM\_ERROR                 = 1  
\*   OTHERS                        = 2  
          .  
if sy-subrc <> 0.  
\* Implement suitable error handling here  
endif.  
  
  
  
  
  
  
endform.



FINAL OUTPUT:



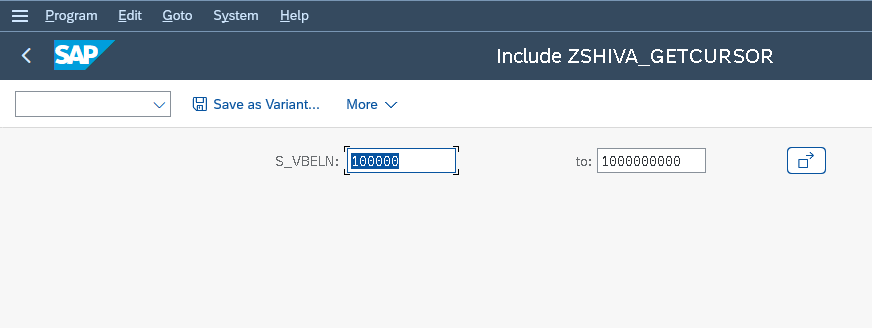
\*&---------------------------------------------------------------------\*  
\*& Report ZSHIVA\_C\_RAISE\_EXC  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSHIVA\_C\_RAISE\_EXC.  
  
CLASS C1 DEFINITION.  
  PUBLIC SECTION.  
  METHODS: CONSTRUCTOR IMPORTING NUM TYPE I  
                       EXCEPTIONS E1.  
ENDCLASS.  
  
CLASS C1 IMPLEMENTATION.  
  METHOD : CONSTRUCTOR.  
    IF NUM LT 7.  
      RAISE E1.  
    ELSE.  
      WRITE :/5 NUM.  
    ENDIF.  
  ENDMETHOD.  
ENDCLASS.  
  
PARAMETERS P\_NO TYPE I .  
  
START-OF-SELECTION.  
DATA : OBJ TYPE REF TO C1.  
CREATE OBJECT : OBJ EXPORTING NUM = P\_NO  
                    EXCEPTIONS E1 = 4.  
  
IF SY-SUBRC = 4.  
  WRITE :/5 'EXCEPTION RADISED PLEASE ENTER NUM GREATEN THAN 7'.  
  ENDIF.



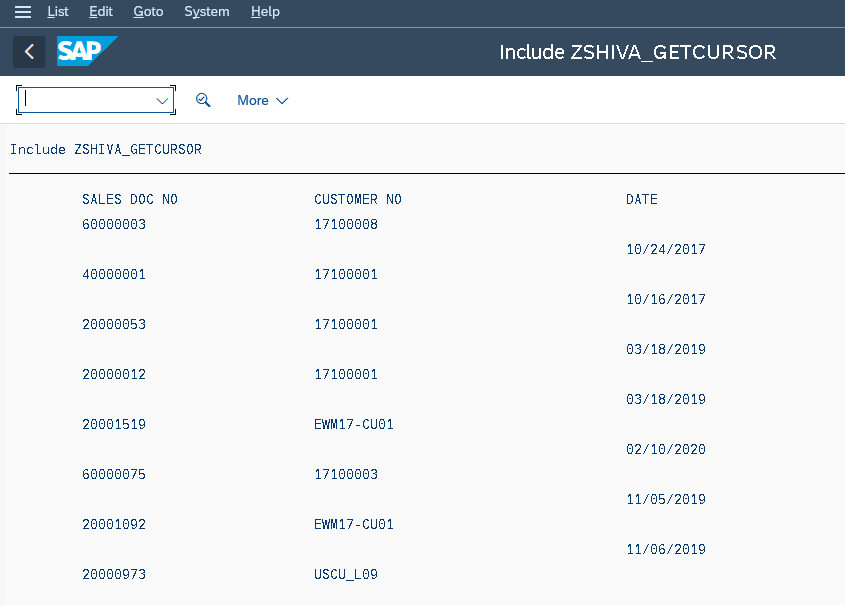
FINAL OUTPUT:



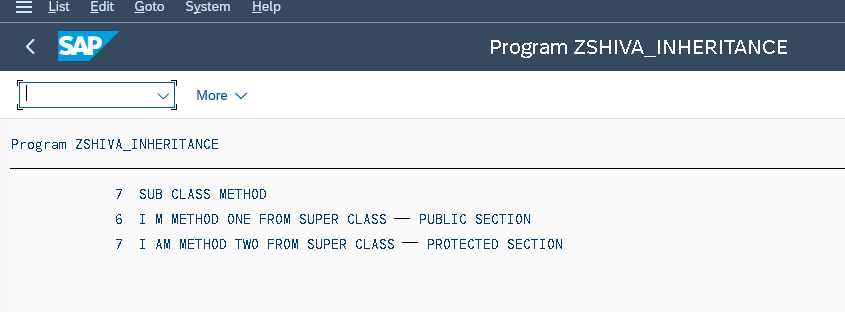
\*&---------------------------------------------------------------------\*  
\*& Report ZSHIVA\_GETCURSOR  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSHIVA\_GETCURSOR.  
TYPES : BEGIN OF TY\_VBAK,  
        VBELN TYPE VBELN\_VA,  
        KUNNR TYPE KUNNR,  
        ERDAT TYPE ERDAT,  
END OF TY\_VBAK.  
DATA : IT\_VBAK TYPE TABLE OF TY\_VBAK,  
      WA\_VBAK TYPE TY\_VBAK.  
  
DATA : LV\_VBELN TYPE VBELN\_VA.  
  
SELECT-OPTIONS S\_VBELN FOR LV\_VBELN.  
START-OF-SELECTION.  
     SELECT VBELN KUNNR ERDAT INTO TABLE IT\_VBAK FROM VBAK WHERE VBELN IN S\_VBELN.  
END-OF-SELECTION.  
  
LOOP AT IT\_VBAK INTO WA\_VBAK.  
  FORMAT HOTSPOT ON.  
  WRITE :/10 WA\_VBAK-VBELN,  
          39 WA\_VBAK-KUNNR.  
  FORMAT HOTSPOT OFF.  
  WRITE :/78 WA\_VBAK-ERDAT.  
ENDLOOP.  
  
TOP-OF-PAGE.  
WRITE :/10 'SALES DOC NO',  
          39 'CUSTOMER NO',  
          78 'DATE'.  
AT LINE-SELECTION.  
  DATA : FNAM(15),FVAL(15).  
  
  GET CURSOR FIELD FNAM VALUE FVAL.  
  IF FNAM = 'WA\_VBAK-VBELN'.  
    SET PARAMETER ID 'AUN' FIELD FVAL.  
    CALL TRANSACTION 'VA03' AND SKIP FIRST SCREEN.  
    ELSEIF FNAM = 'WA\_VBAK-KUNNR'.  
      SET PARAMETER ID 'KUN' FIELD FVAL.  
      CALL TRANSACTION 'XD02' AND SKIP FIRST SCREEN.  
  ENDIF.



FINAL OUTPUT:

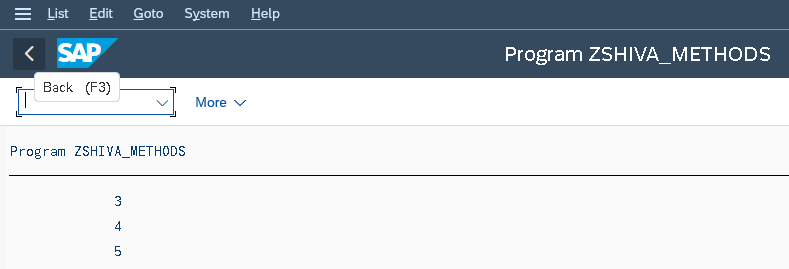


\* SUB CLASS CAN ACCCESS PUBLIC/PROTECTED COMPONENTS OF SUPER CLASS  
REPORT ZSHIVA\_INHERITANCE.  
  
CLASS C1 DEFINITION.  
  PUBLIC SECTION.  
  METHODS: METHOD1.  
  DATA : NUM1 TYPE I VALUE 6.  
  
  PROTECTED SECTION.  
  METHODS : METHOD2.  
  DATA : NUM2 TYPE I VALUE 7.  
ENDCLASS.  
  
CLASS C1 IMPLEMENTATION.  
  METHOD : METHOD1.  
    WRITE :/5 NUM1 ,'I M METHOD ONE FROM SUPER CLASS -- PUBLIC SECTION'.  
    "WRITE :/5 NUM2,'I AM' .  
   ENDMETHOD.  
   METHOD : METHOD2.  
     WRITE :/5 NUM2,'I AM METHOD TWO FROM SUPER CLASS -- PROTECTED SECTION'.  
     ENDMETHOD.  
ENDCLASS.  
CLASS C2 DEFINITION INHERITING FROM C1.  
  PUBLIC SECTION.  
  METHODS : SUBCLASS\_METHOD.  
ENDCLASS.  
  
CLASS C2 IMPLEMENTATION.  
  METHOD : SUBCLASS\_METHOD.  
  
    WRITE :/5 NUM2,'SUB CLASS METHOD'.  
    CALL METHOD : METHOD1, METHOD2.  
  ENDMETHOD.  
ENDCLASS.  
  
  
START-OF-SELECTION.  
DATA OBJ TYPE REF TO C2.  
CREATE OBJECT  OBJ.  
OBJ->SUBCLASS\_METHOD( ).  " ONE WAY  
  
"CALL METHOD OBJ->SUBCLASS\_METHOD.  " SECOND WAY.



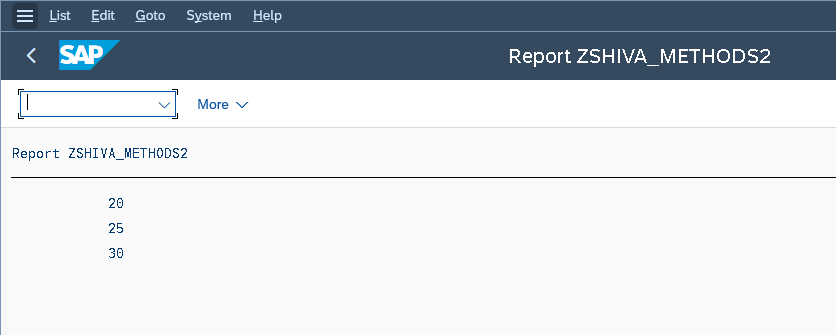
\*&---------------------------------------------------------------------\*  
\*& Report ZSHIVA\_METHODS  
\*\*# 1 Use OF STATIC Attributes  
REPORT ZSHIVA\_METHODS.  
CLASS C1 DEFINITION.  
PUBLIC SECTION.  
CLASS-DATA : NUM TYPE I.  "STATIC ATTRIBUTE.  
METHODS : M1.  " INSTANCE METHOD.  
ENDCLASS.  
  
CLASS C1 IMPLEMENTATION.  
  METHOD M1.  
    NUM = NUM + 1.  
    WRITE :/5 NUM.  
  ENDMETHOD.  
ENDCLASS.  
  
START-OF-SELECTION.  
C1=>NUM = 3.  
WRITE:/5 C1=>NUM.  
  
DATA: OBJ1 TYPE REF TO C1,  
      OBJ2 TYPE REF TO C1.  
CREATE OBJECT : OBJ1,  
                OBJ2.  
  
CALL METHOD OBJ1->M1.  
CALL METHOD OBJ2->M1.

OUTPUT:



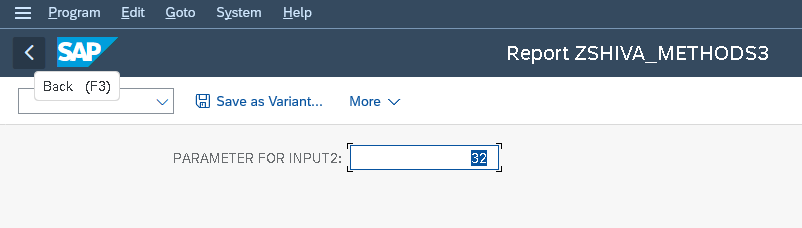
\*&---------------------------------------------------------------------\*  
\*& Report ZSHIVA\_METHODS2  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSHIVA\_METHODS2.  
\*# 2 Different ways OF calling the METHOD WITH one IMPORT PARAMETER.  
CLASS C1 DEFINITION.  
PUBLIC SECTION.  
DATA: NUM TYPE I VALUE 5." INSTANCE ATTRIBUTE ,BECAUSE WE ARE NOT USING CLASS-DATA,[STATIC].  
METHODS: M1 IMPORTING I\_INPUT TYPE I.  " INSTANCE METHOD.  
ENDCLASS.  
  
CLASS C1 IMPLEMENTATION.  
  METHOD:  M1.  
    NUM = NUM \* I\_INPUT.  
    WRITE:/5 NUM.  
    NUM = 5.  
  ENDMETHOD.  
ENDCLASS.  
  
START-OF-SELECTION.  
DATA: OBJ1 TYPE REF TO C1.  
  
CREATE OBJECT : OBJ1.  
CALL METHOD OBJ1->M1 EXPORTING I\_INPUT = 4.  
CALL METHOD OBJ1->M1( I\_INPUT = 5 ).  
OBJ1->M1( 6 ).

OUTPUT :

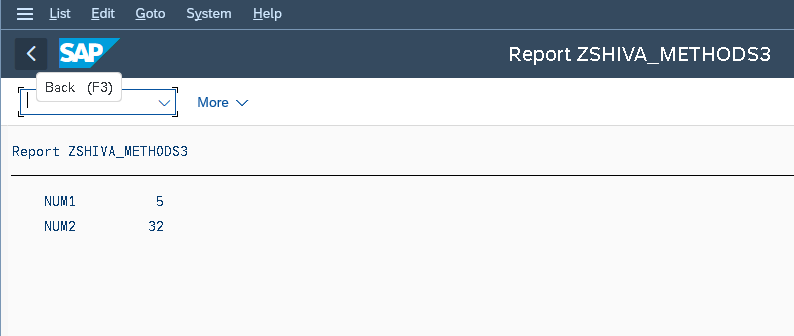


\*&---------------------------------------------------------------------\*  
\*& Report ZSHIVA\_METHODS3  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSHIVA\_METHODS3.  
\*\*# 3  IMPORT PARAMETERS passed BY REF. can’t be changed inside the METHOD  
PARAMETERS P\_NUM TYPE I.  
DATA : NUM TYPE I.  
CLASS C1 DEFINITION.  
  PUBLIC SECTION.  
  METHODS : M1 IMPORTING INPUT1 TYPE I " BY REF.  
                         VALUE(INPUT2) TYPE I. " BY VALUE  
ENDCLASS.  
CLASS C1 IMPLEMENTATION.  
  METHOD: M1.  
    INPUT2 = P\_NUM.  
    "INPUT1 = P\_NUM. " THIS FILED CANNOT BE MODIFIED BECAUSE IT IS REFERENCE TYPE  
                     " IMPORT PARAMETRERS PASSED BY REF CANNOT BE MODIFIED INSIDE METHOD  
    WRITE :/5 'NUM1', INPUT1.  
    WRITE :/5 'NUM2', INPUT2.  
  ENDMETHOD.  
ENDCLASS.  
START-OF-SELECTION.  
DATA: OBJ TYPE REF TO C1.  
CREATE OBJECT : OBJ.  
CALL METHOD OBJ->M1 EXPORTING INPUT1 = 5  
                              INPUT2 = P\_NUM.

OUTPUT:

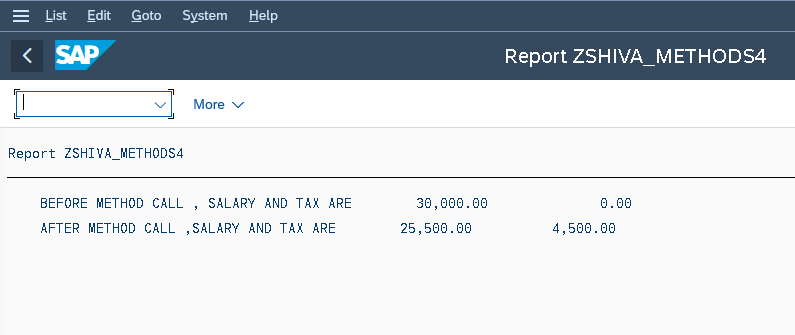


FINAL OUTPUT:

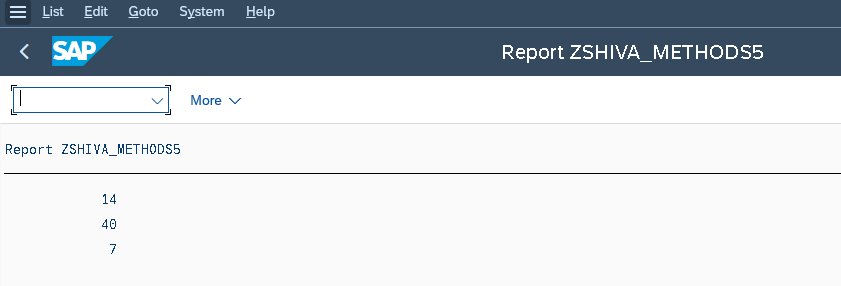


\*&---------------------------------------------------------------------\*  
\*& Report ZSHIVA\_METHODS4  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSHIVA\_METHODS4.  
\* USE OF EXPORT AND CHANGING PARAMETERS OF AMETHOD.  
  
PARAMETERS P\_A(10) TYPE C." GRADES - A01,A02,A03--  
DATA: LV\_TAX TYPE P DECIMALS 2,  
      LV\_SALARY TYPE P DECIMALS 2.  
  
CLASS C1 DEFINITION.  
  PUBLIC SECTION.  
  METHODS : TAX\_CAL IMPORTING I\_GRADE TYPE C  
                    EXPORTING E\_ITAX TYPE P  
                    CHANGING SALARY TYPE P.  
ENDCLASS.  
  
CLASS C1 IMPLEMENTATION.  
  METHOD: TAX\_CAL.  
    CASE I\_GRADE.  
      WHEN 'AO1'.  
        E\_ITAX = SALARY \* '0.2'.  
      WHEN 'AO2'.  
        E\_ITAX = SALARY \* '0.1'.  
      WHEN OTHERS.  
        E\_ITAX = SALARY \* '0.15'.  
       ENDCASE.  
  
       SALARY = SALARY - E\_ITAX.  
   ENDMETHOD.  
ENDCLASS.  
  
START-OF-SELECTION.  
  
DATA : OBJ TYPE REF TO C1.  
CREATE OBJECT : OBJ.  
  
LV\_SALARY = 30000.  
LV\_TAX = 0.  
  
WRITE :/5 'BEFORE METHOD CALL , SALARY AND TAX ARE',  
            LV\_SALARY ,  
            LV\_TAX.  
CALL METHOD OBJ->TAX\_CAL  
EXPORTING  
  I\_GRADE = P\_A " SELECT THE OPTIONS A01  
IMPORTING  
  E\_ITAX = LV\_TAX  
CHANGING  
  SALARY = LV\_SALARY.  
  
WRITE :/5 'AFTER METHOD CALL ,SALARY AND TAX ARE',  
                                LV\_SALARY,  
                                LV\_TAX.

OUTPUT:

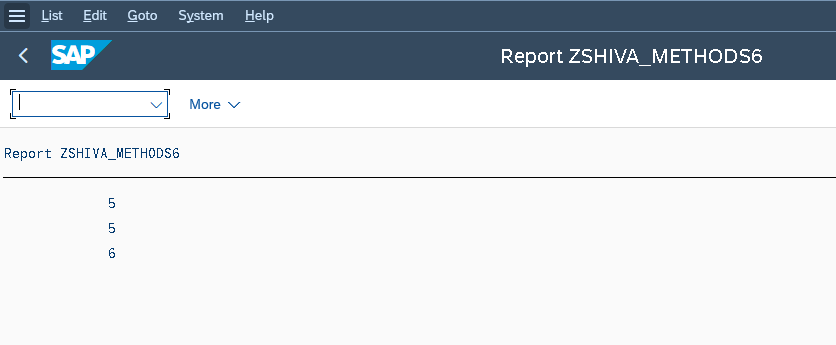


REPORT ZSHIVA\_METHODS5.  
\* USE OF RETURNING PARAMETRES IN METHOD (FUNCTION METHOD)  
DATA: WA\_NUM TYPE I.  
  
CLASS C1 DEFINITION.  
  PUBLIC SECTION.  
  METHODS : M1 IMPORTING INPUT1  TYPE I  
                          INPUT2 TYPE I  
                          RETURNING VALUE(RESULT) TYPE I.  
  ENDCLASS.  
  
  CLASS C1 IMPLEMENTATION.  
     METHOD : M1.  
       RESULT = INPUT1 \* 2 + INPUT2.  
     ENDMETHOD.  
 ENDCLASS.  
  
 START-OF-SELECTION.  
 DATA : OBJ TYPE REF TO C1.  
 CREATE OBJECT : OBJ.  
  
 CALL METHOD OBJ->M1 EXPORTING INPUT1 = 5  
                               INPUT2 = 4  
                               RECEIVING RESULT = WA\_NUM.  
 WRITE :/5 WA\_NUM.  
  
 WA\_NUM = OBJ->M1( INPUT1 = 10 INPUT2 = 20 ).  
 WRITE :/5 WA\_NUM.  
  
 MOVE OBJ->M1( INPUT1 = 2 INPUT2 = 3 ) TO WA\_NUM.  
 WRITE :/5 WA\_NUM.

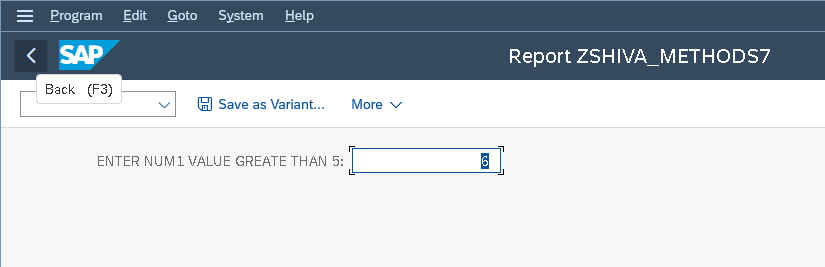


\*STATIC METHOD CAN USE ONLY USE FOR STATIC ATTRIBUTE ,  
\* WE CANNOT CALL/USE INSTANCE ATTRIBUTE IN STATIC METHOD  
REPORT ZSHIVA\_METHODS6.  
  
CLASS C1 DEFINITION.  
  PUBLIC SECTION.  
  CLASS-DATA : STATIC\_NUM TYPE I VALUE 5 ." STATIC ATTRIBUTE.  
  DATA       : INSTANCE\_NUM TYPE I VALUE 6 ."INSTANCE ATTRIBUTE.  
  CLASS-METHODS : STATIC\_M1. " STATIC METHOD  
  METHODS : INSTANCE\_M1.       " INSTANCE METHOD.  
ENDCLASS.  
  
CLASS C1 IMPLEMENTATION.  
  METHOD :  STATIC\_M1  .  
    WRITE :/5 STATIC\_NUM.  
\*    WRITE :/5 INSTANCE\_NUM." THIS IS INSTANCE ATTRIBUTE SO WE CANNT CALL BY USING STATIC  
  ENDMETHOD.  
  
  METHOD : INSTANCE\_M1.  
    WRITE :/5 STATIC\_NUM.  
    WRITE :/5 INSTANCE\_NUM.  
  ENDMETHOD.  
ENDCLASS.  
  
START-OF-SELECTION.  
DATA : OBJ TYPE REF TO C1.  
CALL METHOD C1=>STATIC\_M1.  
  
CREATE OBJECT OBJ.  
CALL METHOD OBJ->INSTANCE\_M1.  
\* OUTPUT-->5  5   6.

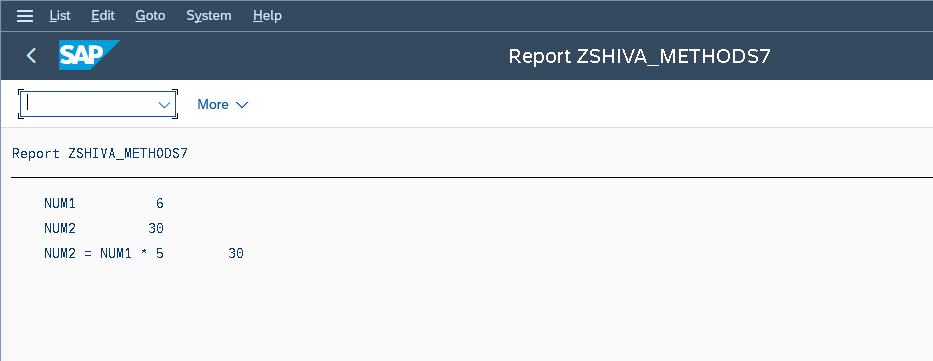
OUTPUT:



\*METHOD RAISING EXCEPTIONS  
REPORT ZSHIVA\_METHODS7.  
  
CLASS C1 DEFINITION.  
  PUBLIC SECTION.  
  METHODS : M1 IMPORTING NUM1 TYPE I " INSTANCE METHOD  
               EXPORTING NUM2 TYPE I  
               EXCEPTIONS E1.  
ENDCLASS.  
CLASS C1 IMPLEMENTATION.  
  METHOD : M1.  
    IF NUM1 LT 5. " IF NUMBER LESS THAN 5.  
      MESSAGE I398(00) WITH 'SHOULD BE GRATERE THAN [>=]5 ' RAISING E1.  
      ELSE.  
         WRITE :/5 'NUM1',10 NUM1.  
        NUM2 = NUM1 \* 5.  
  
           WRITE :/5 'NUM2',10 NUM2.  
  
     ENDIF.  
  ENDMETHOD.  
ENDCLASS.  
PARAMETERS : P\_NO TYPE I.  
START-OF-SELECTION.  
DATA: OBJ TYPE REF TO C1.  
CREATE OBJECT : OBJ.  
  
CALL METHOD  
OBJ->M1 EXPORTING NUM1 = P\_NO  
        IMPORTING NUM2 = P\_NO  
        EXCEPTIONS E1 = 4.  
  
  
IF SY-SUBRC = 4.  
  MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO  
  WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.  
 ELSE.  
   WRITE :/5 'NUM2 = NUM1 \* 5' ,20 P\_NO.  
  
ENDIF.



OUTPUT:



\*&---------------------------------------------------------------------\*  
\*& Report ZSHI\_INT\_REPO\_USERCMD  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSHI\_INT\_REPO\_USERCMD no standard page heading line-count 20(3)  
line-size 500 message-id ZSHIVA\_EROR\_MSG.  
  
load-of-program.  
  
include ZSHI\_ATUSERCMND\_INC.  
  
SELECT-OPTIONS S\_VBELN FOR WA\_TAB1-VBELN OBLIGATORY.  
  
INITIALIZATION.  
CLEAR : WA\_TAB1, LT\_TAB1.  
S\_VBELN-LOW = '1'.  
S\_VBELN-HIGH = '100'.  
  
APPEND S\_VBELN.  
  
AT SELECTION-SCREEN ON S\_VBELN.  
  
  SELECT SINGLE VBELN  
    INTO LV\_VBELN  
    FROM VBAP  
    WHERE VBELN IN S\_VBELN.  
  
IF SY-SUBRC EQ 0.  
  MESSAGE S000.  
ELSE.  
  MESSAGE E001.  
ENDIF.  
  
START-OF-SELECTION.  
  
SET PF-STATUS 'ZSHIVA\_PFS' EXCLUDING 'INFO'.  
  
PERFORM GET\_VBAP\_DATA.  
  
END-OF-SELECTION.  
PERFORM DISP\_VBAP\_DATA.  
  
DATA LV\_COUNT TYPE I.  
LV\_COUNT = SY-LINCT - SY-LINNO.  
SKIP LV\_COUNT.  
  
TOP-OF-PAGE.  
 ULINE.  
 WRITE :/ 'RECORDS ARE FOUND IN VBAP TABLE', 'THE CURRENT LIST INDEX IS :', SY-LSIND.  
 WRITE :/ 'SALES DOCUMENT',  
       30 'MATERIAL NO',  
       42 'MATERIAL ENT',  
       57 'ITEM CATEGORY',  
       76 'SALES ORDER ITEM'.  
 ULINE.  
 END-OF-PAGE.  
 WRITE :/ 'CURRENT LIST PAGENO : ',SY-PAGNO,  
          'DATE:' , SY-DATUM,  
          'TIME:', SY-UZEIT.  
  
AT USER-COMMAND.  
  
  CASE SY-UCOMM.  
    WHEN 'DISP'.  
      WRITE 'DISPLAY INFO'.  
    WHEN 'INFO'.  
      WRITE /: 'THE CURRENT DATE:', SY-DATUM,  
               'TIMEL', SY-UZEIT.  
  
    WHEN 'TCODE'.  
      CALL TRANSACTION 'ZRAS\_VBAP\_TCODE'.  
  
    WHEN 'WELCOME TO SAP'.  
      WRITE 'SAP  UST GLOBAL'.  
  
    WHEN 'ABAP'.  
      WRITE 'SAP ABAP HYD'.  
   ENDCASE.  
  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_VBAP\_DATA  
\*&---------------------------------------------------------------------\*  
\*& text---------------\*  
FORM get\_vbap\_data .  
  
SELECT VBELN MATNR MATWA PSTYV ARKTX  
  INTO TABLE LT\_TAB1  
  FROM VBAP  
  WHERE VBELN IN S\_VBELN.  
  
  IF SY-SUBRC EQ 0.  
    WRITE :/ 'RECORDS ARE FOUND IN VBAP TABLE'.  
   ELSE.  
     WRITE :/ 'RECORDS ARE NOT FOUND IN TABLE'.  
  ENDIF.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form DISP\_VBAP\_DATA  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM disp\_vbap\_data .  
 LOOP AT LT\_TAB1 INTO WA\_TAB1.  
   WRITE :/ WA\_TAB1-VBELN HOTSPOT,  
         30 WA\_TAB1-MATNR,  
         42 WA\_TAB1-MATWA,  
         57 WA\_TAB1-PSTYV,  
         76 WA\_TAB1-ARKTX.  
  
   HIDE WA\_TAB1-VBELN.  
 ENDLOOP.  
  
ENDFORM.

OUTPUT:

