1. **VALIDATIONS AND OPERATIONS AT SELECTION SCREEN LEVEL:**

REPORT Z357\_DAY11\_SELSCR NO STANDARD PAGE HEADING LINE-COUNT 20(2) LINE-SIZE 500 MESSAGE-ID Z357\_CLASS\_MYMESSAGE.  
  
TYPES: BEGIN OF LS\_TAB,  
        MATNR TYPE MATNR, "MATERIAL NUMBER  
        MTART TYPE MTART, "MATERIAL TYPE  
        MBRSH TYPE MBRSH, "INDUSTRY TYPE  
        MEINS TYPE MEINS, "UNIT OF MEASURE  
  
      END OF LS\_TAB.  
  
DATA : LT\_TAB TYPE STANDARD TABLE OF LS\_TAB,  
        WA\_TAB TYPE LS\_TAB,  
        LV\_MATNR TYPE MATNR.  
  
SELECT-OPTIONS S\_MATNR FOR WA\_TAB-MATNR OBLIGATORY NO-EXTENSION.  
  
SELECTION-SCREEN : BEGIN OF BLOCK B1 WITH FRAME TITLE TEXT-001.  
PARAMETERS: R1 RADIOBUTTON GROUP RG1 USER-COMMAND FLAG DEFAULT 'X',  
            R2 RADIOBUTTON GROUP RG1 .  
  
SELECTION-SCREEN ULINE /10(60).  
  
SELECTION-SCREEN : END OF BLOCK B1.  
  
  
SELECTION-SCREEN : BEGIN OF BLOCK B2 WITH FRAME TITLE TEXT-002.  
  
SELECTION-SCREEN ULINE /10(60).  
SELECTION-SCREEN SKIP.  
SELECTION-SCREEN COMMENT /30(70) L\_CMT.  
  
SELECTION-SCREEN : END OF BLOCK B2.  
  
  
SELECTION-SCREEN : BEGIN OF BLOCK B3 WITH FRAME TITLE TEXT-003.  
PARAMETERS:P\_MTART TYPE MTART MODIF ID M1,  
           P\_MBRSH TYPE MBRSH MODIF ID M2.  
  
PARAMETERS:  UST\_HYD AS CHECKBOX DEFAULT 'X',  
             UST\_US AS CHECKBOX.  
  
SELECTION-SCREEN : END OF BLOCK B3.  
  
\*INITIALIZATION.  
\*        CLEAR: WA\_TAB , LT\_TAB.  
\*  
\*        L\_CMT = 'UST -HYD'.  
\*        S\_MATNR-LOW = '1'.  
\*        S\_MATNR-HIGH = '100'.  
\*        P\_MTART = 'FERT'.  
\*        APPEND S\_MATNR.  
  
\*AT SELECTION-SCREEN ON S\_MATNR.  
\*  SELECT SINGLE MATNR INTO LV\_MATNR FROM MARA WHERE MATNR IN S\_MATNR.  
\*    IF sy-subrc EQ 0.  
\*      MESSAGE s000.  
\*    ELSE.  
\*      MESSAGE e001.  
\*    ENDIF.  
  
AT SELECTION-SCREEN OUTPUT.  
LOOP AT SCREEN.  
  IF R1 = 'X' AND  
    SCREEN-GROUP1 = 'M2'.  
      SCREEN-ACTIVE = '0'.  
   ELSEIF R2 = 'X' AND  
     SCREEN-GROUP1 = 'M1'.  
      SCREEN-ACTIVE = '0'.  
   ENDIF.  
   MODIFY SCREEN.  
  ENDLOOP.  
  
  
IF R1 = 'X'.  
     PERFORM GET\_MTART.  
  ELSE.  
     PERFORM GET\_MBRSH.  
ENDIF.  
  
  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_MTART  
\*&---------------------------------------------------------------------\*  
  
FORM get\_mtart .  
SELECT MATNR MTART MBRSH MEINS  
      INTO TABLE LT\_TAB  
      FROM MARA  
      WHERE MATNR IN S\_MATNR AND  
      MTART EQ P\_MTART.  
  
LOOP AT LT\_TAB INTO WA\_TAB.  
    WRITE: / WA\_TAB-MATNR,  
             WA\_TAB-MTART,  
             WA\_TAB-MBRSH,  
             WA\_TAB-MEINS.  
 ENDLOOP.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_MBRSH  
\*&---------------------------------------------------------------------\*  
\*& text  
  
FORM get\_mbrsh .  
SELECT MATNR MTART MBRSH MEINS  
      INTO TABLE LT\_TAB  
      FROM MARA  
      WHERE MATNR IN S\_MATNR AND  
      MTART EQ P\_MTART.  
  
LOOP AT LT\_TAB INTO WA\_TAB.  
    WRITE: / WA\_TAB-MATNR,  
             WA\_TAB-MTART,  
             WA\_TAB-MBRSH,  
             WA\_TAB-MEINS.  
ENDLOOP.  
ENDFORM.

1. **CREATING A SIMPLE INTERACTIVE REPORT WITH A SECONDARY LIST**

REPORT Z280357\_DAY9\_IR1 NO STANDARD PAGE HEADING LINE-COUNT 20(3) LINE-SIZE 500.  
  
LOAD-OF-PROGRAM.  
\*INCLUDE Z280357\_DAY7\_INC\_TOP.  
  
TYPES : BEGIN OF LS\_TAB,  
          KUNNR TYPE KUNNR,  
          LAND1 TYPE LAND1\_GP,  
          NAME1 TYPE NAME1\_GP,  
          ORT01 TYPE ORT01\_GP,  
        END OF LS\_TAB,  
  
        BEGIN OF LS\_TAB2,  
           KUNNR TYPE KUNNR,  
           LOEVM TYPE LOEVM\_B,  
          SPERR TYPE SPERB\_X,  
        END OF LS\_TAB2.  
  
DATA : LT\_TAB TYPE STANDARD TABLE OF LS\_TAB,  
      WA\_TAB TYPE LS\_TAB,  
      LV\_KUNNR TYPE KUNNR,  
  
      LT\_TAB2 TYPE STANDARD TABLE OF LS\_TAB2,  
      WA\_TAB2 TYPE LS\_TAB2.  
  
  
  
SELECT-OPTIONS S\_KUNNR FOR LV\_KUNNR OBLIGATORY.  
  
INITIALIZATION.  
  
CLEAR: LT\_TAB, WA\_TAB.  
  
S\_KUNNR-LOW = '0001000143'.  
S\_KUNNR-HIGH = '0001000200'.  
  
APPEND S\_KUNNR.  
  
AT SELECTION-SCREEN ON S\_KUNNR.  
  SELECT SINGLE KUNNR INTO LV\_KUNNR FROM KNA1 WHERE KUNNR IN S\_KUNNR.  
  IF SY-SUBRC EQ 0.  
     MESSAGE S000(Z357\_CLASS\_MYMESSAGE).  
     ELSE.  
      MESSAGE E001(Z357\_CLASS\_MYMESSAGE).  
  ENDIF.  
START-OF-SELECTION.  
PERFORM GET\_CUST\_MAST.  
END-OF-SELECTION.  
  
PERFORM DISP\_CUST\_MAST.  
DATA LV\_COUNT TYPE I.  
LV\_COUNT = SY-LINCT - SY-LINNO.  
SKIP LV\_COUNT.  
  
TOP-OF-PAGE.  
ULINE.  
WRITE : / 'KUNNR' ,  
          10 'LAND1',  
          20 'NAME1',  
          35 'ORT01'.  
ULINE.  
END-OF-PAGE.  
WRITE: / 'CURRENT PAGE NO. IS ' , SY-PAGNO,  
           'DATE: ' , SY-DATUM,  
           'TIME: ' , SY-UZEIT.  
  
AT LINE-SELECTION.  
  CASE SY-LSIND.  
    WHEN 1.  
      SELECT KUNNR SPERR LOEVM  
        INTO TABLE LT\_TAB2  
        FROM KNB1  
        WHERE KUNNR EQ WA\_TAB-KUNNR.  
\*        WRITE 'HELLLO'.  
      LOOP AT LT\_TAB2 INTO WA\_TAB2.  
        WRITE : / WA\_TAB2-KUNNR HOTSPOT,  
                  WA\_TAB2-SPERR,  
                  WA\_TAB2-LOEVM.  
        HIDE WA\_TAB2-KUNNR.  
      ENDLOOP.  
  ENDCASE.  
  
\*&---------------------------------------------------------------------\*  
FORM get\_cust\_mast .  
  SELECT KUNNR LAND1 NAME1 ORT01  
    INTO TABLE LT\_TAB  
    FROM KNA1  
    WHERE KUNNR IN S\_KUNNR.  
  if sy-subrc eq 0.  
    write :/ 'Records are found in mara table'.  
  else.  
    write : / 'Records are not found in mara table'.  
  endif.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
FORM disp\_cust\_mast .  
LOOP AT LT\_TAB INTO WA\_TAB.  
    WRITE : / WA\_TAB-KUNNR HOTSPOT,  
             10 WA\_TAB-LAND1,  
             20 WA\_TAB-NAME1,  
             35 WA\_TAB-ORT01.  
    HIDE WA\_TAB-KUNNR.  
    ENDLOOP.  
ENDFORM.

1. **IMPORTANT KEYWORDS IN INTERNAL TABLES CONCEPTS**

REPORT Z357\_DAY11\_ITAB\_KEYWORDS.  
  
DATA : BEGIN OF LINE, "FIELD STRING.  
        COL1(10) TYPE C,  
        COL2(20) TYPE C,  
  END OF LINE.  
  
DATA : LT\_TAB LIKE TABLE OF LINE.  
1ST LINE-STRING FIELD DECLARATION ADN USAGE\*\*\*\*\*  
LINE-COL1 = 'US'.  
LINE-COL2 = 'TECHNOLOGIES'.  
APPEND LINE TO LT\_TAB.  
  
IF LT\_TAB IS INITIAL.  
   WRITE 'INTERNAL TABLE IS EMPTY'.  
   FREE LT\_TAB.  
   ELSE.  
     WRITE : LINE-COL1 , LINE-COL2.  
ENDIF.  
  
  
\*USING INSERT AND TABINDEX FOR ROW SELECTION OF NEW ENTRIES.\*\*\*\*\*  
DATA : BEGIN OF LINE,  
        NAME(20) TYPE C,  
        AGE TYPE I,  
        WEIGHT TYPE P DECIMALS 2,  
        LAND(10) TYPE C,  
        END OF LINE.  
DATA : LT\_TAB LIKE SORTED TABLE OF LINE WITH NON-UNIQUE KEY NAME AGE WEIGHT LAND.  
  
LINE-NAME = 'SHIK'.  
LINE-AGE = '20'.  
LINE-WEIGHT = '45.50'.  
LINE-LAND = 'NZB'.  
INSERT LINE INTO TABLE LT\_TAB.  
CLEAR LINE.  
  
LINE-NAME = 'SHAK'.  
LINE-AGE = '19'.  
LINE-WEIGHT = '38.9'.  
LINE-LAND = 'HYD'.  
INSERT LINE INTO TABLE LT\_TAB.  
CLEAR LINE.  
  
LINE-NAME = 'SHOO'.  
LINE-AGE = '22'.  
LINE-WEIGHT = '54.12'.  
LINE-LAND = 'MNR'.  
INSERT LINE INTO TABLE LT\_TAB.  
CLEAR LINE.  
  
  
LOOP AT LT\_TAB INTO LINE.  
  WRITE: / LINE-NAME , LINE-AGE , LINE-WEIGHT , LINE-LAND.  
  
  ENDLOOP.  
  
  
\*\*\*\*\*\*\*\* #3 INSERT LINE\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
DATA: BEGIN OF LINE,  " fld strng  
  col\_1 TYPE I,  
  col\_2 TYPE I,  
END OF LINE.  
  
DATA itab LIKE TABLE OF LINE.  
  
  
DO 2 TIMES.  
  LINE-col\_1 = sy-INDEX.  
  LINE-col\_2 = sy-INDEX \*\* 2.  
  APPEND LINE TO itab.  
ENDDO.  
  
LOOP  AT itab INTO LINE.  
  LINE-col\_1 = 3 \* sy-tabix.  
  LINE-col\_2 = 5 \* sy-tabix.  
  INSERT LINE INTO itab.  
ENDLOOP.  
LOOP AT itab INTO LINE.  
  WRITE: / sy-tabix, LINE-col\_1, LINE-col\_2.  
ENDLOOP.  
  
  
\*\*\*MODIFY INTL TABLE DATA THROUGH LINE\*\*\*\*\*\*\*\*  
DATA: BEGIN OF LINE,  
  col1 TYPE I,  
  col2 TYPE I,  
END OF LINE.  
  
DATA itab LIKE TABLE OF LINE.  
  
DO 3 TIMES.  
  LINE-col1 = sy-INDEX.  
  LINE-col2 = sy-INDEX \*\* 2.  
  APPEND LINE TO itab.  
ENDDO.  
  
LOOP AT itab INTO LINE.  
  IF sy-tabix = 2.  
    LINE-col1 = sy-tabix  \*  10.  
    LINE-col2 = ( sy-tabix  \* 10 ) \*\* 2.  
    MODIFY itab FROM LINE.  
  ENDIF.  
ENDLOOP.  
LOOP AT itab INTO LINE.  
  WRITE: / sy-tabix, LINE-col1, LINE-col2.  
ENDLOOP.  
  
DELETE OPERATION ON ITAB\*\*\*\*\*\*  
DATA: BEGIN OF LINE,  
  col1 TYPE I,  
  col2 TYPE I,  
END OF LINE.  
DATA itab LIKE HASHED TABLE OF LINE WITH UNIQUE key col1.  
  
DO 4 TIMES.  
  LINE-col1 = sy-INDEX.  
  LINE-col2 = sy-INDEX \*\* 2.  
  INSERT LINE INTO TABLE itab.  
ENDDO.  
LINE-col1 = 1.  
DELETE TABLE itab: FROM LINE, WITH TABLE KEY col1 = 3.  
  
LOOP AT itab INTO LINE.  
  WRITE: / LINE-col1, LINE-col2.  
ENDLOOP.  
  
  
  
  
\*\*\*\*\*\*\*\*\*COLLECT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
DATA: BEGIN OF LINE,  
  col\_1(3) TYPE C,  
  col\_2(2) TYPE n,  
  col\_3    TYPE I,  
END OF LINE.  
  
DATA itab LIKE SORTED TABLE OF LINE  
      WITH NON-UNIQUE KEY col\_1  col\_2.  
  
LINE-col\_1 = 'UST'.  
LINE-col\_2 = '12'.  
LINE-col\_3 = 3.  
  
COLLECT LINE INTO itab.  
WRITE / sy-tabix.  
  
  
LINE-col\_1 = 'TECH'.  
LINE-col\_2 = '34'.  
LINE-col\_3 = 5.  
COLLECT LINE INTO itab.  
WRITE / sy-tabix.  
  
  
LINE-col\_1 = 'UST'.  
LINE-col\_2 = '12'.  
LINE-col\_3 = 7.  
COLLECT LINE INTO itab.  
  
WRITE / sy-tabix.  
LOOP AT itab INTO LINE.  
  WRITE: / LINE-col\_1, LINE-col\_2, LINE-col\_3.  
ENDLOOP.

1. **CREATING A “GRID” DISPLAY ALV REPORT USING PRE-DEFINED ALV TOOLS.**

REPORT Z357\_DAY12\_ALV\_GRID\_DISP.  
  
LOAD-OF-PROGRAM.  
INCLUDE Z357\_DATA\_DEC\_INCLUDE\_TOP.  
  
SELECT-OPTIONS S\_KUNNR FOR LV\_KUNNR OBLIGATORY.  
  
INITIALIZATION.  
  
CLEAR: LT\_TAB, WA\_FIELDCAT, LT\_FIELDCAT.  
  
S\_KUNNR-LOW = '0001000143'.  
S\_KUNNR-HIGH = '0001000200'.  
APPEND S\_KUNNR.  
  
  
AT SELECTION-SCREEN ON S\_KUNNR.  
  SELECT SINGLE KUNNR INTO LV\_KUNNR FROM KNA1 WHERE KUNNR IN S\_KUNNR.  
  
  IF SY-SUBRC EQ 0.  
     MESSAGE S000(Z357\_CLASS\_MYMESSAGE).  
     ELSE.  
      MESSAGE E001(Z357\_CLASS\_MYMESSAGE).  
  ENDIF.  
  
  
START-OF-SELECTION.  
  PERFORM GET\_ALV\_FCAT.  
  
  PERFORM GET\_CUST\_MAST.  
  
END-OF-SELECTION.  
  
PERFORM ALV\_GRID\_DISP.  
  
FORM get\_cust\_mast .  
  SELECT KUNNR LAND1 NAME1 ORT01 PSTLZ TELF1  
    INTO TABLE LT\_TAB  
    FROM KNA1  
    WHERE KUNNR IN S\_KUNNR.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALV\_FCAT  
\*&---------------------------------------------------------------------\*  
\*& text  
FORM get\_alv\_fcat .  
  WA\_FIELDCAT-COL\_POS = '1'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Customer No.'.  
  WA\_FIELDCAT-FIELDNAME = 'KUNNR'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '2'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Region/Country'.  
  WA\_FIELDCAT-FIELDNAME = 'LAND1'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '3'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Name Of Cust'.  
  WA\_FIELDCAT-FIELDNAME = 'NAME1'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '4'.  
  WA\_FIELDCAT-SELTEXT\_L = 'City Of Cust'.  
  WA\_FIELDCAT-FIELDNAME = 'ORT01'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '5'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Postal Code'.  
  WA\_FIELDCAT-FIELDNAME = 'PSTLZ'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '6'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Telephone'.  
  WA\_FIELDCAT-FIELDNAME = 'TELF1'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form ALV\_GRID\_DISP  
\*&---------------------------------------------------------------------\*  
\*& text  
FORM alv\_grid\_disp .  
  CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
   EXPORTING  
     I\_CALLBACK\_PROGRAM                = SY-REPID  
     I\_GRID\_TITLE                      = 'Cust Master Information'  
\*     I\_GRID\_SETTINGS                   =  
\*     IS\_LAYOUT                         =  
     IT\_FIELDCAT                       = LT\_FIELDCAT  
  
    TABLES  
      t\_outtab                          = LT\_TAB  
   EXCEPTIONS  
     PROGRAM\_ERROR                     = 1  
     OTHERS                            = 2  
            .  
ENDFORM.

1. **CREATING A “GRID” DISPLAY ALV REPORT ALONG WITH A “LOGO”.**

REPORT Z357\_DAY12\_ALV\_GRID\_DISP\_LOGO.  
  
LOAD-OF-PROGRAM.  
INCLUDE Z357\_DATA\_DEC\_INCLUDE\_TOP.  
  
SELECT-OPTIONS S\_KUNNR FOR LV\_KUNNR OBLIGATORY.  
  
INITIALIZATION.  
  
CLEAR: LT\_TAB, WA\_FIELDCAT, LT\_FIELDCAT.  
  
S\_KUNNR-LOW = '0001000143'.  
S\_KUNNR-HIGH = '0001000200'.  
APPEND S\_KUNNR.  
  
  
AT SELECTION-SCREEN ON S\_KUNNR.  
  SELECT SINGLE KUNNR INTO LV\_KUNNR FROM KNA1 WHERE KUNNR IN S\_KUNNR.  
  
  
  IF SY-SUBRC EQ 0.  
     MESSAGE S000(Z357\_CLASS\_MYMESSAGE).  
     ELSE.  
      MESSAGE E001(Z357\_CLASS\_MYMESSAGE).  
  ENDIF.  
  
  
START-OF-SELECTION.  
\*  PERFORM GET\_ALV\_FCAT.  
  PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING TEXT-001 'KUNNR' CS\_TAB 'C100'.  
  PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING TEXT-002 'LAND1' CS\_TAB 'C200'.  
  PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING TEXT-003 'NAME1' CS\_TAB 'C300'.  
  PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING TEXT-004 'ORT01' CS\_TAB 'C400'.  
  PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING TEXT-005 'PSTLZ' CS\_TAB 'C500'.  
  PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING TEXT-006 'TELF1' CS\_TAB 'C600'.  
  
  PERFORM GET\_CUST\_MAST.  
END-OF-SELECTION.  
  
  
PERFORM ALV\_GRID\_DISP.  
  
FORM get\_cust\_mast .  
  SELECT KUNNR LAND1 NAME1 ORT01 PSTLZ TELF1  
    INTO TABLE LT\_TAB  
    FROM KNA1  
    WHERE KUNNR IN S\_KUNNR.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALV\_FCAT  
\*&---------------------------------------------------------------------\*  
\*& text  
FORM get\_alv\_fcat .  
  WA\_FIELDCAT-COL\_POS = '1'.  
  WA\_FIELDCAT-SELTEXT\_L = TEXT-001.  
  WA\_FIELDCAT-FIELDNAME = 'KUNNR'.  
  WA\_FIELDCAT-TABNAME = cs\_tab.  
  WA\_FIELDCAT-no\_zero = 'X'.  
  WA\_FIELDCAT-emphasize = 'C100'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '2'.  
  WA\_FIELDCAT-SELTEXT\_L = TEXT-002.  
  WA\_FIELDCAT-FIELDNAME = LAND1.  
  WA\_FIELDCAT-TABNAME = cs\_tab.  
  append wa\_fieldcat to lt\_fieldcat.  
  WA\_FIELDCAT-emphasize = 'C600'.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '3'.  
  WA\_FIELDCAT-SELTEXT\_L = TEXT-003.  
  WA\_FIELDCAT-FIELDNAME = NAME.  
  WA\_FIELDCAT-TABNAME = cs\_tab.  
  WA\_FIELDCAT-no\_zero = 'X'.  
  WA\_FIELDCAT-emphasize = 'C500'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '4'.  
  WA\_FIELDCAT-SELTEXT\_L = TEXT-004.  
  WA\_FIELDCAT-FIELDNAME = 'ORT01'.  
  WA\_FIELDCAT-TABNAME = cs\_tab.  
  WA\_FIELDCAT-no\_zero = 'X'.  
  WA\_FIELDCAT-emphasize = 'C400'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '5'.  
  WA\_FIELDCAT-SELTEXT\_L = TEXT-005.  
  WA\_FIELDCAT-FIELDNAME = 'PSTLZ'.  
  WA\_FIELDCAT-TABNAME = cs\_tab.  
  WA\_FIELDCAT-no\_zero = 'X'.  
  WA\_FIELDCAT-emphasize = 'C300'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '6'.  
  WA\_FIELDCAT-SELTEXT\_L = TEXT-006.  
  WA\_FIELDCAT-FIELDNAME = 'TELF1'.  
  WA\_FIELDCAT-TABNAME = cs\_tab.  
  WA\_FIELDCAT-no\_zero = 'X'.  
  WA\_FIELDCAT-emphasize = 'C200'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form ALV\_GRID\_DISP  
\*&---------------------------------------------------------------------\*  
\*& text  
FORM alv\_grid\_disp .  
\*  TY\_LAYOUT-no\_colhead = 'X'."WHY ARE THESE NOT WORKING?????  
\*  TY\_LAYOUT-zebra = 'X'.  
\*  ty\_layout-no\_vline = 'X' .  
\*  ty\_layout-no\_hline = 'X'.  
  
  CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
   EXPORTING  
     I\_CALLBACK\_PROGRAM                = SY-REPID  
     I\_CALLBACK\_TOP\_OF\_PAGE            = 'GETLOGO' "CREATE SUBROUTINE TO INCLUDE LOGO AT TOP  
\*     I\_CALLBACK\_HTML\_TOP\_OF\_PAGE       = ' '  
     I\_GRID\_TITLE                      = LV\_TITLE  
\*     I\_GRID\_SETTINGS                   =  
\*     IS\_LAYOUT                         = TY\_LAYOUT  
     IT\_FIELDCAT                       = LT\_FIELDCAT  
    TABLES  
      t\_outtab                          = LT\_TAB  
   EXCEPTIONS  
     PROGRAM\_ERROR                     = 1  
     OTHERS                            = 2  
            .  
ENDFORM.  
  
FORM GETLOGO.  
  DATA : WA\_HEADER TYPE SLIS\_LISTHEADER,  
         LT\_HEADER TYPE SLIS\_T\_LISTHEADER.  
  
  WA\_HEADER-typ = 'H'.  
  WA\_HEADER-info = 'UST GLOBAL'.  
  WA\_HEADER-key = 'HYDERABAD'. "NO DISPLAY IN HEADER  
  APPEND WA\_HEADER TO LT\_HEADER.  
  CLEAR WA\_HEADER.  
  
  WA\_HEADER-typ = 'S'.  
  WA\_HEADER-info = 'ABAP'.  
  WA\_HEADER-key = 'SAP'. "DISPLAY KEY IN SELECTION TYPE  
  APPEND WA\_HEADER TO LT\_HEADER.  
  CLEAR WA\_HEADER.  
  
  WA\_HEADER-typ = 'A'.  
  WA\_HEADER-info = SY-DATUM.  
  WA\_HEADER-key = SY-UZEIT. "NO DISPLAY IN ACTION TYP  
  APPEND WA\_HEADER TO LT\_HEADER.  
  CLEAR WA\_HEADER.  
  
  CALL FUNCTION 'REUSE\_ALV\_COMMENTARY\_WRITE'  
    EXPORTING  
      it\_list\_commentary       = LT\_HEADER  
      I\_LOGO                   = 'Z357\_DAY12\_ALV\_GRID\_DISPLAY\_LOGO'  
\*     I\_END\_OF\_LIST\_GRID       =  
\*     I\_ALV\_FORM               =  
            .  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALV\_FCAT\_USING\_PERFORM  
\*&---------------------------------------------------------------------\*  
FORM get\_alv\_fcat\_using\_perform USING P\_A P\_B P\_C P\_D.  
  WA\_FIELDCAT-COL\_POS = LV\_COUNT.  
  WA\_FIELDCAT-SELTEXT\_L = P\_A.  
  WA\_FIELDCAT-FIELDNAME = P\_B.  
  WA\_FIELDCAT-TABNAME = P\_C.  
  WA\_FIELDCAT-no\_zero = 'X'.  
  WA\_FIELDCAT-emphasize = P\_D.  
  
  LV\_COUNT = LV\_COUNT + 1.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
ENDFORM.

1. **CREATING A SIMPLE “LIST” DISPLAY ALV REPORT**

REPORT Z357\_DAY12\_ALV\_LIST\_DISP.  
  
LOAD-OF-PROGRAM.  
INCLUDE Z357\_DATA\_DEC\_INCLUDE\_TOP.  
  
SELECT-OPTIONS S\_KUNNR FOR LV\_KUNNR OBLIGATORY.  
  
INITIALIZATION.  
  
CLEAR: LT\_TAB, WA\_FIELDCAT, LT\_FIELDCAT.  
  
S\_KUNNR-LOW = '0001000143'.  
S\_KUNNR-HIGH = '0001000200'.  
APPEND S\_KUNNR.  
  
  
AT SELECTION-SCREEN ON S\_KUNNR.  
  SELECT SINGLE KUNNR INTO LV\_KUNNR FROM KNA1 WHERE KUNNR IN S\_KUNNR.  
  
  IF SY-SUBRC EQ 0.  
     MESSAGE S000(Z357\_CLASS\_MYMESSAGE).  
     ELSE.  
      MESSAGE E001(Z357\_CLASS\_MYMESSAGE).  
  ENDIF.  
  
  
START-OF-SELECTION.  
  PERFORM GET\_ALV\_FCAT.  
  
  PERFORM GET\_CUST\_MAST.  
  
END-OF-SELECTION.  
  
perform alv\_list\_ltfc\_op.  
  
FORM get\_cust\_mast .  
  SELECT KUNNR LAND1 NAME1 ORT01  
    INTO TABLE LT\_TAB  
    FROM KNA1  
    WHERE KUNNR IN S\_KUNNR.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALV\_FCAT  
\*&---------------------------------------------------------------------\*  
\*& text  
FORM get\_alv\_fcat .  
  WA\_FIELDCAT-COL\_POS = '1'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Customer No.'.  
  WA\_FIELDCAT-FIELDNAME = 'KUNNR'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '2'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Region/Country'.  
  WA\_FIELDCAT-FIELDNAME = 'LAND1'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '3'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Name Of Cust'.  
  WA\_FIELDCAT-FIELDNAME = 'NAME1'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '4'.  
  WA\_FIELDCAT-SELTEXT\_L = 'City Of Cust'.  
  WA\_FIELDCAT-FIELDNAME = 'ORT01'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '5'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Postal Code'.  
  WA\_FIELDCAT-FIELDNAME = 'PSTLZ'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '6'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Telephone'.  
  WA\_FIELDCAT-FIELDNAME = 'TELF1'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form alv\_list\_ltfc\_op  
\*&---------------------------------------------------------------------\*  
  
FORM alv\_list\_ltfc\_op .  
  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  
    TABLES  
      t\_outtab                       = LT\_TAB  
   EXCEPTIONS  
     PROGRAM\_ERROR                  = 1  
     OTHERS                         = 2  
            .  
  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
  ENDIF.  
  
ENDFORM.

1. **CREATING A “POPUP” DISPLAY REPORT USING ALV TOOLS.**

REPORT Z357\_DAY12\_ALV\_POPUP\_DISP.  
  
LOAD-OF-PROGRAM.  
INCLUDE Z357\_DATA\_DEC\_INCLUDE\_TOP.  
  
SELECT-OPTIONS S\_KUNNR FOR LV\_KUNNR OBLIGATORY.  
  
INITIALIZATION.  
  
CLEAR: LT\_TAB, WA\_FIELDCAT, LT\_FIELDCAT.  
  
S\_KUNNR-LOW = '0001000143'.  
S\_KUNNR-HIGH = '0001000200'.  
APPEND S\_KUNNR.  
  
  
AT SELECTION-SCREEN ON S\_KUNNR.  
  SELECT SINGLE KUNNR INTO LV\_KUNNR FROM KNA1 WHERE KUNNR IN S\_KUNNR.  
  
  IF SY-SUBRC EQ 0.  
     MESSAGE S000(Z357\_CLASS\_MYMESSAGE).  
     ELSE.  
      MESSAGE E001(Z357\_CLASS\_MYMESSAGE).  
  ENDIF.  
  
  
START-OF-SELECTION.  
  PERFORM GET\_ALV\_FCAT.  
  
  PERFORM GET\_CUST\_MAST.  
  
END-OF-SELECTION.  
  
PERFORM ALV\_POPUP\_DISP.  
  
FORM get\_cust\_mast .  
  SELECT KUNNR LAND1 NAME1 ORT01 PSTLZ TELF1  
    INTO TABLE LT\_TAB  
    FROM KNA1  
    WHERE KUNNR IN S\_KUNNR.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALV\_FCAT  
\*&---------------------------------------------------------------------\*  
\*& text  
FORM get\_alv\_fcat .  
  WA\_FIELDCAT-COL\_POS = '1'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Customer No.'.  
  WA\_FIELDCAT-FIELDNAME = 'KUNNR'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '2'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Region/Country'.  
  WA\_FIELDCAT-FIELDNAME = 'LAND1'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '3'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Name Of Cust'.  
  WA\_FIELDCAT-FIELDNAME = 'NAME1'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '4'.  
  WA\_FIELDCAT-SELTEXT\_L = 'City Of Cust'.  
  WA\_FIELDCAT-FIELDNAME = 'ORT01'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '5'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Postal Code'.  
  WA\_FIELDCAT-FIELDNAME = 'PSTLZ'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  WA\_FIELDCAT-COL\_POS = '6'.  
  WA\_FIELDCAT-SELTEXT\_L = 'Telephone'.  
  WA\_FIELDCAT-FIELDNAME = 'TELF1'.  
  WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form ALV\_POPUP\_DISP  
\*&---------------------------------------------------------------------\*  
\*& text  
FORM alv\_popup\_disp .  
  CALL FUNCTION 'REUSE\_ALV\_POPUP\_TO\_SELECT'  
    EXPORTING  
     I\_TITLE                       = 'Customer Master Information'  
\*     I\_SELECTION                   = 'X'  
\*     I\_ALLOW\_NO\_SELECTION          =  
     I\_ZEBRA                       = 'X'  
      i\_tabname                     = 'LT\_TAB'  
     I\_STRUCTURE\_NAME              = 'LS\_TAB'  
     IT\_FIELDCAT                   = LT\_FIELDCAT  
\*     IT\_EXCLUDING                  =  
     I\_CALLBACK\_PROGRAM            = SY-REPID  
    tables  
      t\_outtab                      = LT\_TAB  
   EXCEPTIONS  
     PROGRAM\_ERROR                 = 1  
     OTHERS                        = 2  
            .  
  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
  ENDIF.  
  
ENDFORM.

1. **CREATING AN** **HIERARCHIAL - SEQUENTIAL REPORT USING ALV TOOLS:**

REPORT Z357\_DAY13\_ALV\_HIERSEQ\_LIST.  
  
INCLUDE Z357\_DATA\_DEC\_INCLUDE\_TOP.  
  
TABLES VBAK.  
  
SELECT-OPTIONS S\_VBELN FOR VBAK-VBELN.  
  
INITIALIZATION.  
CLEAR: WA\_VBAK, WA\_VBAP , LT\_VBAK, LT\_VBAP, WA\_FIELDCAT, LT\_FIELDCAT.  
  
S\_VBELN-LOW = '5000'.  
S\_VBELN-HIGH = '5010'.  
APPEND S\_VBELN.  
  
START-OF-SELECTION.  
  
PERFORM GET\_DATA.  
\*PERFORM GET\_ALV\_FCAT. NOT USING HARDCODE PERFORM  
PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING 'VBELN' 'LT\_VBAK' 'VBAK' 'VBELN' 1.  
PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING 'ERDAT' 'LT\_VBAK' 'VBAK' 'ERDAT' 2.  
PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING 'ERZET' 'LT\_VBAK' 'VBAK' 'ERZET' 3.  
  
  
\*CATALOG FOR THE SEQUENTIAL DATA  
PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING 'VBELN' 'LT\_VBAP' 'VBAP' 'VBELN' 1.  
PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING 'POSNR' 'LT\_VBAP' 'VBAP' 'POSNR' 2.  
PERFORM GET\_ALV\_FCAT\_USING\_PERFORM USING 'MATNR' 'LT\_VBAP' 'VBAP' 'MATNR' 3.  
  
END-OF-SELECTION.  
  
PERFORM HIER\_DISPLAY.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_DATA  
\*&---------------------------------------------------------------------\*  
\*& text  
FORM get\_data .  
  SELECT VBELN ERDAT ERZET  
    INTO CORRESPONDING FIELDS OF TABLE LT\_VBAK  
    FROM VBAK  
    WHERE VBELN IN S\_VBELN.  
  
  SELECT VBELN POSNR MATNR  
    INTO TABLE LT\_VBAP  
    FROM VBAP  
    WHERE VBELN IN S\_VBELN.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALV\_FCAT\_USING\_PERFORM  
\*&---------------------------------------------------------------------\*  
\*& text  
FORM get\_alv\_fcat\_using\_perform  USING P\_A P\_B P\_C P\_D P\_F.  
  WA\_FIELDCAT-COL\_POS = P\_F.  
\*  WA\_FIELDCAT-SELTEXT\_L = P\_A.  
  WA\_FIELDCAT-FIELDNAME = P\_A.  
  WA\_FIELDCAT-TABNAME = P\_B.  
  WA\_FIELDCAT-REF\_TABNAME = P\_C.  
  WA\_FIELDCAT-REF\_FIELDNAME = P\_D.  
  WA\_FIELDCAT-no\_zero = 'X'.  
\*  WA\_FIELDCAT-emphasize = P\_E.  
  
\*  LV\_COUNT = LV\_COUNT + 1.  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form HIER\_DISPLAY  
\*&---------------------------------------------------------------------\*  
\*&  
FORM hier\_display .  
  IT\_KEY-HEADER01 = 'VBELN'.  
  TY\_LAYOUT-expand\_fieldname = 'CH'.  
  IF SY-SUBRC EQ 0.  
    CALL FUNCTION 'REUSE\_ALV\_HIERSEQ\_LIST\_DISPLAY'  
      EXPORTING  
\*       I\_INTERFACE\_CHECK              = ' '  
       I\_CALLBACK\_PROGRAM             = SY-REPID  
\*       I\_CALLBACK\_PF\_STATUS\_SET       = ' '  
\*       I\_CALLBACK\_USER\_COMMAND        = ' '  
       IS\_LAYOUT                      = TY\_LAYOUT  
       IT\_FIELDCAT                    = LT\_FIELDCAT[]  
\*       i\_tabname\_header               = 'LT\_VBAK'  
        i\_tabname\_item                 = 'LT\_VBAP'  
\*       I\_STRUCTURE\_NAME\_HEADER        =  
\*       I\_STRUCTURE\_NAME\_ITEM          =  
        is\_keyinfo                     = IT\_KEY  
\*       TABLES  
        t\_outtab\_header                = LT\_VBAK  
        t\_outtab\_item                  = LT\_VBAP  
     EXCEPTIONS  
       PROGRAM\_ERROR                  = 1  
       OTHERS                         = 2  
              .  
    IF sy-subrc <> 0.  
\* Implement suitable error handling here  
    ENDIF.  
  
  ENDIF.  
ENDFORM.

1. **JOINING OR POOLING TWO TABLE DATA INTO ONE LIST REPORT USING “FOR ALL ENTRIES”:**

REPORT Z357\_DAY15\_FORALLENTRIES.  
  
INCLUDE Z357\_INCLUDE\_FORALLENTRIES\_TOP.  
  
  
SELECT-OPTIONS S\_VBELN FOR LV\_VBELN.  
  
INITIALIZATION.  
       CLEAR: LT\_TAB1, WA\_TAB1, LT\_TAB2, WA\_TAB2.  
       S\_VBELN-LOW = '1'.  
       S\_VBELN-HIGH = '100'.  
       APPEND S\_VBELN.  
  
AT SELECTION-SCREEN ON S\_VBELN.  
  SELECT SINGLE VBELN INTO LV\_VBELN FROM VBAK WHERE VBELN IN S\_VBELN.  
  IF SY-SUBRC EQ 0.  
    MESSAGE S000(Z357\_CLASS\_MYMESSAGE).  
    ELSE.  
     MESSAGE E001(Z357\_CLASS\_MYMESSAGE).  
  ENDIF.  
  
START-OF-SELECTION.  
  SELECT VBELN ERDAT ERZET ERNAM AUDAT  
    INTO TABLE LT\_TAB1  
    FROM VBAK  
    WHERE VBELN IN S\_VBELN.  
  
    IF NOT LT\_TAB1[] IS INITIAL.  
      SELECT VBELN POSNR MATNR MATWA  
        INTO TABLE LT\_TAB2  
        FROM VBAP  
        FOR ALL ENTRIES IN LT\_TAB1  
        WHERE VBELN EQ LT\_TAB1-VBELN.  
  
        DELETE ADJACENT DUPLICATES FROM LT\_TAB1 COMPARING VBELN.  
  
        SORT LT\_TAB2 BY VBELN.  
  
        LOOP AT LT\_TAB1 INTO WA\_TAB1.  
          READ TABLE LT\_TAB2 INTO WA\_TAB2 WITH KEY VBELN = WA\_TAB1-VBELN BINARY SEARCH.  
  
          IF SY-SUBRC EQ 0.  
            MOVE : WA\_TAB1-VBELN TO WA\_FINALTAB-VBELN,  
                   WA\_TAB1-ERDAT TO WA\_FINALTAB-ERDAT,  
                   WA\_TAB1-ERZET TO WA\_FINALTAB-ERZET,  
                   WA\_TAB1-ERNAM TO WA\_FINALTAB-ERNAM,  
                   WA\_TAB1-AUDAT TO WA\_FINALTAB-AUDAT,  
  
                   WA\_TAB2-POSNR TO WA\_FINALTAB-POSNR,  
                   WA\_TAB2-MATNR TO WA\_FINALTAB-MATNR,  
                   WA\_TAB2-MATWA TO WA\_FINALTAB-MATWA  
                   .  
            APPEND WA\_FINALTAB TO LT\_FINALTAB.  
  
            ELSE.  
              CONTINUE.  
          ENDIF.  
  
        ENDLOOP.  
  
        LOOP AT LT\_FINALTAB INTO WA\_FINALTAB.  
          WRITE: / WA\_FINALTAB-VBELN,  
                   WA\_FINALTAB-ERDAT,  
                   WA\_FINALTAB-ERZET,  
                   WA\_FINALTAB-ERNAM,  
                   WA\_FINALTAB-AUDAT,  
                   WA\_FINALTAB-POSNR,  
                   WA\_FINALTAB-MATNR,  
                   WA\_FINALTAB-MATWA.  
        ENDLOOP.  
      ELSE.  
        WRITE / 'NO RECORDS FOUND...'.  
    ENDIF.

1. **POOLING**

REPORT Z357\_DAY15\_IR\_GETCURSOR.  
INCLUDE Z357\_INCLUDE\_IR\_GETCURSOR\_TOP.

\*INCLUDES DATA

\*TYPES : BEGIN OF LS\_vbak,   "FIELDS FROM VBAK   
\*\*          vbeln TYPE vbeln\_va,  
\*\*          kunnr TYPE kunnr,  
\*\*          erdat TYPE erdat,  
\*\*        END OF LS\_vbak.  
\*\*  
\*\*DATA :  LT\_vbak TYPE TABLE OF LS\_vbak,  
\*\*        wa\_vbak TYPE LS\_vbak,  
\*\*        lv\_vbeln type vbeln\_va.  
  
  
\*PROGRAM

\*\*\*\*# 1  
\*\*START-OF-SELECTION.  
\*\*  
\*\*FORMAT COLOR 7.  
\*\*WRITE / 'BASIC LIST'.  
\*\*  
\*\*FORMAT HOTSPOT ON.  
\*\*WRITE: / VAR1, VAR2, VAR3.  
\*\*FORMAT HOTSPOT OFF.  
\*\*  
\*\*SKIP 3.  
\*\*WRITE / 'UST@HYDERABAD'.  
\*\*  
\*\*AT LINE-SELECTION.  
\*\*  
\*\*GET CURSOR FIELD FNAME.  
\*\*  
\*\*IF FNAME = 'VAR1'.  
\*\*  WRITE 'im VAR1'.  
\*\*  ELSEIF FNAME = 'VAR2'.  
\*\*    WRITE 'IM VAR2 INTERACTION'.  
\*\*  ELSEIF FNAME = 'VAR3'.  
\*\*    WRITE 'IM VAR3 INTN'.  
\*\*ENDIF.  
  
  
SELECT-OPTIONS S\_VBELN FOR LV\_VBELN.  
  
START-OF-SELECTION.  
SELECT VBELN KUNNR ERDAT  
  INTO TABLE LT\_VBAK  
  FROM VBAK  
  WHERE VBELN IN S\_VBELN.  
  
END-OF-SELECTION.  
  
LOOP AT LT\_VBAK INTO WA\_VBAK.  
  FORMAT HOTSPOT ON.  
  WRITE : / WA\_VBAK-VBELN,  
            WA\_VBAK-KUNNR.  
  FORMAT HOTSPOT OFF.  
  WRITE WA\_VBAK-ERDAT.  
ENDLOOP.  
  
top-OF-PAGE.  
WRITE : / 'SaleS doc No',  
'Customerno',  
'Date'.  
  
 AT LINE-SELECTION.  
    DATA : FNAME(20),  
          FVAL(20).  
    GET CURSOR FIELD FNAME VALUE FVAL.  
    IF FNAME = 'WA\_VBAK-VBELN'.  
      SET PARAMETER ID 'AUN' FIELD FVAL.  
      CALL TRANSACTION 'VA03' AND SKIP FIRST SCREEN.  
  
      ELSEIF FNAME = 'WA\_VBAK-KUNNR'.  
        SET PARAMETER ID 'KUN' FIELD FVAL.  
        CALL TRANSACTION 'XD02' AND SKIP FIRST SCREEN.  
ENDIF.

1. **OBJECT ORIENTED ABAP BASICS**

**DECLARING METHODS AND ATTRIBUTES**

**CLASS CONSTRUCTORS.**

REPORT Z357\_DAY16\_METHODS\_EXAMPLES.  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*# 1 Use OF STATIC Attributes\*\*8\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*CLASS c1 DEFINITION .  
\*PUBLIC SECTION.  
\*CLASS-DATA : NUM TYPE I .   " STAT ATTR  
\*METHODS : M1. " INST 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,  
\*OBJ2->m1.  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*# 2 Different ways OF calling the METHOD WITH one IMPORT PARAMETER  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*  
\*CLASS C1 DEFINITION.  
\*\*  DATA NUM TYPE I.  
\*  PUBLIC SECTION.  
\*  METHODS: M1 IMPORTING P1 TYPE I  
\*                       VALUE(P2) TYPE I.  
\*  
\*ENDCLASS.  
\*  
\*CLASS C1 IMPLEMENTATION.  
\*  METHOD M1.  
\*\*\*\*    P1 = P1 + 2. "CANNOT DO THIS!!!  
\*    P2 = P2 \* 2.  
\*    WRITE: / P1, P2 .  
\*  ENDMETHOD.  
\*ENDCLASS.  
\*  
\*START-OF-SELECTION.  
\*\*\*APPLICATION  
\*DATA OBJ TYPE REF TO C1.  
\*CREATE OBJECT OBJ.  
\*  
\*PARAMETERS: P\_A TYPE I,  
\*            P\_B TYPE I.  
\*  
\*CALL METHOD OBJ->M1 EXPORTING P1 = P\_A P2 = P\_B.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*# 4 Use OF EXPORT AND CHANGING PARAMETERS OF a METHOD  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*  
\*CLASS C1 DEFINITION.  
\*  PUBLIC SECTION.  
\*  METHODS M1 IMPORTING GRADE TYPE C  
\*             EXPORTING ITAX TYPE P  
\*             CHANGING P\_SALARY TYPE P.  
\*ENDCLASS.  
\*  
\*CLASS C1 IMPLEMENTATION.  
\*  METHOD M1.  
\*    CASE GRADE.  
\*      WHEN 'A01'.  
\*        ITAX = P\_SALARY \* '0.01'.  
\*      WHEN 'A02'.  
\*        ITAX = P\_SALARY \* '0.02'.  
\*      WHEN OTHERS.  
\*        ITAX = P\_SALARY \* '0.015'.  
\*     ENDCASE.  
\*     P\_SALARY = P\_SALARY - ITAX.  
\*  
\*     WRITE : / 'WRITING FROM METHOD, AFTER TAXATION SALARY IS : ' , P\_SALARY.  
\*  ENDMETHOD.  
\*  
\*ENDCLASS.  
\*  
\*START-OF-SELECTION.  
\*\*\*\*APPLICATION  
\*PARAMETERS : P\_GRADE(3) TYPE C,  
\*             P\_SAL TYPE P.  
\*DATA LV\_EXP TYPE P.  
\*DATA OBJ TYPE REF TO C1.  
\*CREATE OBJECT OBJ.  
\*  
\*CALL METHOD OBJ->m1  
\*  EXPORTING  
\*    grade    = 'A02'  
\*  IMPORTING  
\*    itax     = LV\_EXP  
\*  CHANGING  
\*    p\_salary = P\_SAL  
\*  .  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*# 5 USING VALUE RETURN WITH PARAMETERS.  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*  
\*DATA LV\_RES TYPE I.  
\*  
\*CLASS C1 DEFINITION.  
\*  PUBLIC SECTION.  
\*  METHODS M1 IMPORTING I\_A TYPE I I\_B TYPE I  
\*                       RETURNING VALUE(I\_RESULT) TYPE I.  
\*  
\*ENDCLASS.  
\*CLASS C1 IMPLEMENTATION.  
\*  METHOD M1.  
\*    I\_RESULT = I\_A \* 2 + I\_B.  
\*  ENDMETHOD.  
\*ENDCLASS.  
\*  
\*START-OF-SELECTION.  
\*  
\*PARAMETERS: P\_A TYPE I,  
\*            P\_B TYPE I.  
\*  
\*DATA OBJ TYPE REF TO C1.  
\*CREATE OBJECT OBJ.  
\*  
\*DATA LV\_SUM TYPE I.  
\*  
\*LV\_SUM = OBJ->m1(  
\*  EXPORTING  
\*    i\_a      = P\_A  
\*    i\_b      =  P\_B  
\*).  
\*  
\*WRITE /5 LV\_SUM.  
  
  
  
\*\*\*\*\*\*STATIC METHODS can only use STATIC attributes, instance METHODS use both  
\*CLASS C1 DEFINITION.  
\*PUBLIC SECTION.  
\*CLASS-DATA : ST\_X TYPE I VALUE 5.  " STAT ATTR  
\*DATA : IN\_Y TYPE I VALUE 6 .  " INST ATTR  
\*  
\*CLASS-METHODS : ST\_M .   " STAT METH  
\*METHODS : IN\_M .   " INST METY  
\*ENDCLASS.  
\*  
\*CLASS C1 IMPLEMENTATION.  
\*  METHOD : ST\_M .  
\*    WRITE:/5 ST\_X .  
\*\*\*    WRITE:/5 IN\_Y . "CANNOT CALL INSTANCE IN STATIC  
\*  ENDMETHOD.  
\*  
\*  METHOD IN\_M.  
\*    WRITE:/5 ST\_X . "ANYTHING POSSIBLE IN STATIC  
\*    WRITE:/5 IN\_Y .  
\*  ENDMETHOD.  
\*ENDCLASS.  
\*  
\*START-OF-SELECTION.  
\*DATA : OREF1 TYPE REF TO C1.  
\*CALL METHOD c1=>st\_M .  
\*  
\*CREATE OBJECT OREF1.  
\*CALL METHOD oref1->in\_M.  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*# 6.1 INSTANCE CONSTRUCTOR  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*CLASS C1 DEFINITION.  
\*  PUBLIC SECTION.  
\*  METHODS: M1 IMPORTING P\_1 TYPE I,  
\*           CONSTRUCTOR.  
\*ENDCLASS.  
\*  
\*CLASS C1 IMPLEMENTATION.  
\*  METHOD M1.  
\*\*    P\_1 = P\_1 \* 2.  
\*    WRITE /5 P\_1.  
\*  ENDMETHOD.  
\*  
\*  METHOD CONSTRUCTOR.  
\*    WRITE : /5 'I AM IN THE CONSTRUCTOR BLOCK. CALLED BEFORE M1'.  
\*  ENDMETHOD.  
\*ENDCLASS.  
\*  
\*START-OF-SELECTION.  
\*\*APPLICATION  
\*DATA OBJ1 TYPE REF TO C1.  
\*CREATE OBJECT OBJ1.  
\*PARAMETERS P\_A TYPE I.  
\*OBJ1->m1( EXPORTING P\_1 = P\_A ).  
  
  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*# 6.2 INSTANCE CONSTRUCTOR WITH IMPORTS AND EXCEPTIONS  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\*CLASS C1 DEFINITION.  
\*  PUBLIC SECTION.  
\*  METHODS: M1 IMPORTING P\_1 TYPE I,  
\*           CONSTRUCTOR IMPORTING P\_2 TYPE I  
\*                        EXCEPTIONS E.  
\*ENDCLASS.  
\*  
\*CLASS C1 IMPLEMENTATION.  
\*  METHOD M1.  
\*\*    P\_1 = P\_1 \* 2.  
\*    WRITE: /5 'sTATEMENT FROM M1:' , P\_1.  
\*  ENDMETHOD.  
\*  
\*  METHOD CONSTRUCTOR.  
\*    WRITE : /5 'I AM IN THE CONSTRUCTOR BLOCK. CALLED BEFORE M1' , 'P\_2 = ' , P\_2.  
\*    IF P\_2 LT 10.  
\*      RAISE E.  
\*    ENDIF.  
\*  ENDMETHOD.  
\*ENDCLASS.  
\*  
\*START-OF-SELECTION.  
\*  
\*PARAMETERS P\_A TYPE I.  
\*\*APPLICATION  
\*DATA OBJ1 TYPE REF TO C1.  
\*CREATE OBJECT OBJ1 EXPORTING P\_2 = P\_A  
\*                   EXCEPTIONS E = 4.  
\*IF SY-SUBRC EQ 4.  
\*  WRITE /5 'EXCEPTION E RAISED'.  
\*ENDIF.  
\*\*IF E EQ 4.  
\*\*\* MESSAGE ID SY-MSGID TYPE SY-MSGTY NUMBER SY-MSGNO  
\*\*\*   WITH SY-MSGV1 SY-MSGV2 SY-MSGV3 SY-MSGV4.  
\*\* WRITE /5 'EXCEPTION E RAISED'.  
\*\*ENDIF.  
\*OBJ1->m1( EXPORTING P\_1 = P\_A ).