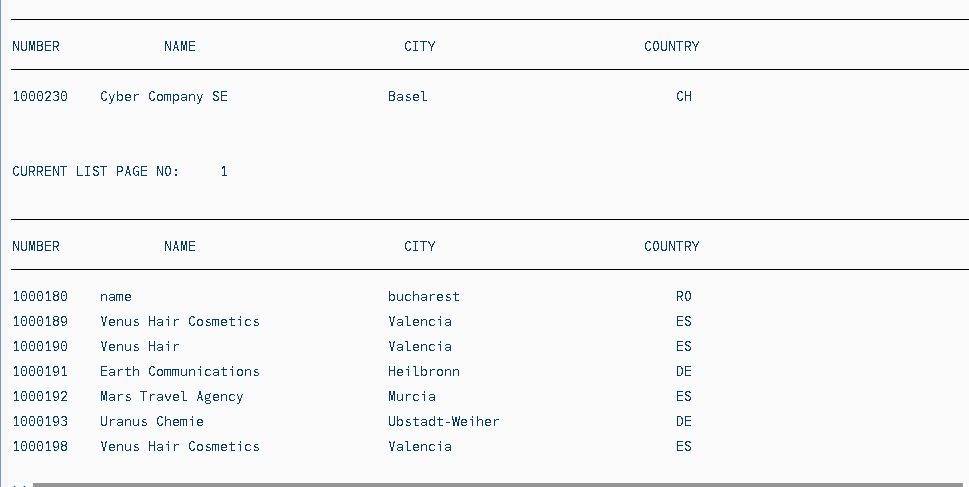
Classical report.

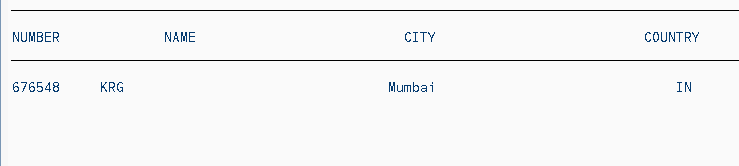
REPORT zsang\_classic NO STANDARD PAGE HEADING LINE-COUNT 20(3) LINE-SIZE 200.  
  
LOAD-OF-PROGRAM.  
  INCLUDE zsang\_classic\_top.  
  
  
  SELECT-OPTIONS s\_kunnr FOR wa\_tab-kunnr OBLIGATORY.  
  
INITIALIZATION.  
  CLEAR: lt\_tab, wa\_tab.  
  s\_kunnr-low = '1000150'.  
  s\_kunnr-high = '1000500'.  
  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(zsang\_msg).  
  ELSE.  
    MESSAGE e001(zsang\_msg).  
  ENDIF.  
  
START-OF-SELECTION.  
set PF-STATUS 'ZSANG\_UST'.  
  PERFORM get\_opt.  
  
END-OF-SELECTION.  
  PERFORM get\_writen.  
  
TOP-OF-PAGE.  
  ULINE.  
  WRITE: / 'NUMBER',  
  20 'NAME',  
  50 'CITY',  
  80 'COUNTRY'.  
  ULINE.  
END-OF-PAGE.  
  SKIP 2.  
  WRITE: / 'CURRENT LIST PAGE NO:', sy-pagno.  
  
  AT LINE-SELECTION.  
    case sy-lsind.  
      when 1.  
        select kunnr bukrs sperr ernam  
          into table lt\_tab1  
          from knb1  
          where kunnr eq wa\_tab-kunnr.  
      endcase.  
      loop at lt\_tab1 into wa\_tab1.  
        write : / wa\_tab1-kunnr HOTSPOT,  
        wa\_tab1-bukrs,  
        wa\_tab1-sperr,  
        wa\_tab1-ernam.  
        ENDLOOP.  
  
at USER-COMMAND.  
  case sy-ucomm.  
    when 'DISP'.  
      WRITE 'DISPLAY INFO'.  
      WHEN 'INFO'.  
        WRITE 'OINFORMATION ABOUT THE TABLE'.  
   ENDCASE.  
  
  
FORM get\_opt .  
  
  SELECT kunnr name1 ort01 land1  
  INTO TABLE lt\_tab  
  FROM kna1  
  WHERE kunnr IN s\_kunnr.  
  
  
ENDFORM.  
  
FORM get\_writen .  
  LOOP AT lt\_tab INTO wa\_tab.  
    WRITE: / wa\_tab-kunnr HOTSPOT,  
    wa\_tab-name1,  
    wa\_tab-ort01,  
    wa\_tab-land1.  
  ENDLOOP.  
  HIDE wa\_tab-kunnr.  
  
ENDFORM.





2. interactive

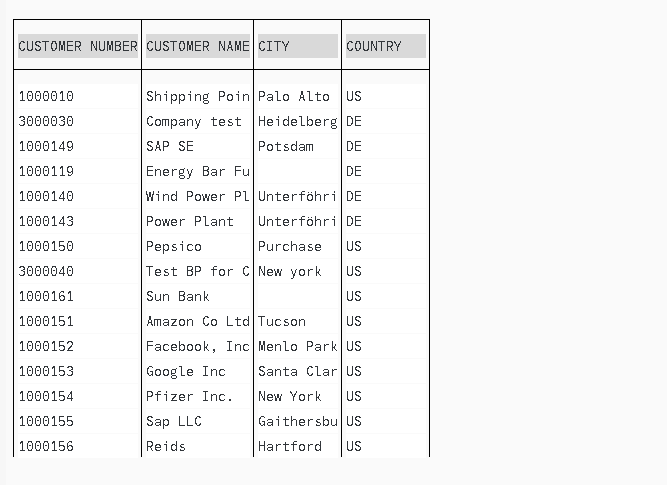
REPORT zsang\_interactive NO STANDARD PAGE HEADING LINE-COUNT 20(3) LINE-SIZE 200.  
  
LOAD-OF-PROGRAM.  
  INCLUDE zsang\_interactive\_top.  
\*  INCLUDE zsang\_classic\_top.  
  
  
  SELECT-OPTIONS s\_kunnr FOR wa\_tab-kunnr OBLIGATORY.  
  
INITIALIZATION.  
  CLEAR: lt\_tab, wa\_tab.  
  s\_kunnr-low = '10'.  
  s\_kunnr-high = '1000000'.  
  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(zsang\_msg).  
  ELSE.  
    MESSAGE e001(zsang\_msg).  
  ENDIF.  
  
START-OF-SELECTION.  
  PERFORM get\_opt.  
  
END-OF-SELECTION.  
  PERFORM get\_writen.  
  
TOP-OF-PAGE.  
  ULINE.  
  WRITE: / 'NUMBER',  
  20 'NAME',  
  50 'CITY',  
  80 'COUNTRY'.  
  ULINE.  
  
END-OF-PAGE.  
  SKIP 2.  
  WRITE: / 'CURRENT LIST PAGE NO:', sy-pagno.  
  
AT LINE-SELECTION.  
  CASE sy-lsind.  
    WHEN 1.  
      SELECT kunnr bukrs sperr ernam  
        INTO TABLE lt\_tab1  
        FROM knb1  
      WHERE kunnr EQ wa\_tab-kunnr.  
  ENDCASE.  
  LOOP AT lt\_tab1 INTO wa\_tab1.  
    WRITE : / wa\_tab1-kunnR,  
    wa\_tab1-bukrs,  
    wa\_tab1-sperr,  
    wa\_tab1-ernam.  
  
  ENDLOOP.  
  
  
  
FORM get\_opt .  
  
  SELECT kunnr name1 ort01 land1  
  INTO TABLE lt\_tab  
  FROM kna1  
  WHERE kunnr IN s\_kunnr.  
  
  
ENDFORM.  
  
FORM get\_writen .  
  LOOP AT lt\_tab INTO wa\_tab.  
    WRITE: / wa\_tab-kunnr HOTSPOT,  
    wa\_tab-name1,  
    wa\_tab-ort01,  
    wa\_tab-land1.  
  ENDLOOP.  
  HIDE wa\_tab-kunnr.  
  
ENDFORM.





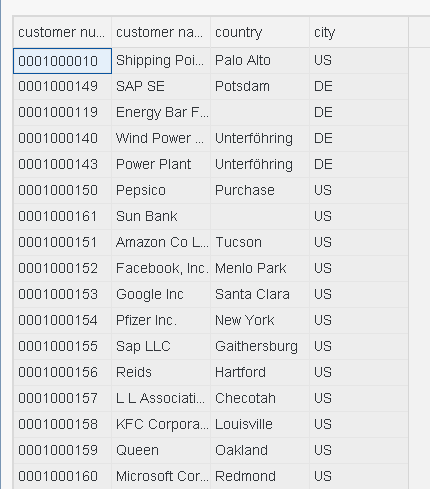
3. ALV LIST

REPORT zsang\_alv\_list\_pro NO STANDARD PAGE HEADING LINE-COUNT 20(3).  
INCLUDE zsang\_alv\_list\_top.  
SELECT-OPTIONS s\_kunnr FOR lv\_kunnr.  
  
INITIALIZATION.  
  CLEAR: lt\_tab, lt\_fieldcat.  
  s\_kunnr-low = '1'. " START OF PROGRAM USED FOR DECLARING DEFAULT, CALCULATION AND CLEARING OF INTERNAL TABLES.  
  s\_kunnr-high = '10000'.  
  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(zsang\_msg).  
  ELSE.  
    MESSAGE s001(zsang\_msg).  
    ENDIF.  
  
START-OF-SELECTION. " USED FOR LOADING THE SQL PROGRAM  
  PERFORM get\_select.  
  PERFORM GET\_DATA.  
  
END-OF-SELECTION.  
PERFORM FUNC\_CALL.  
  
  
  
  
FORM get\_select .  
  SELECT kunnr name1 ort01 land1  
    INTO TABLE lt\_tab  
    FROM kna1  
    WHERE kunnr IN s\_kunnr.  
  
ENDFORM.  
  
FORM get\_data .  
wa\_fieldcat-col\_pos = '1'.  
wa\_fieldcat-fieldname = 'KUNNR'.  
wa\_fieldcat-seltext\_l = 'CUSTOMER NUMBER'.  
WA\_FIELDCAT-tabname = 'LT\_TAB'.  
APPEND WA\_FIELDCAT TO LT\_FIELDCAT.  
  
WA\_FIELDCAT-COL\_POS = '2'.  
WA\_FIELDCAT-SELTEXT\_l = 'CUSTOMER NAME'.  
WA\_FIELDCAT-FIELDNAME = 'NAME1'.  
WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
APPEND WA\_FIELDCAT TO LT\_FIELDCAT.  
  
WA\_FIELDCAT-COL\_POS = '3'.  
WA\_FIELDCAT-SELTEXT\_L = 'CITY'.  
WA\_FIELDCAT-FIELDNAME = 'ORT01'.  
WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
APPEND WA\_FIELDCAT TO LT\_FIELDCAT.  
  
WA\_FIELDCAT-COL\_POS = '4'.  
WA\_FIELDCAT-SELTEXT\_L = 'COUNTRY'.  
WA\_FIELDCAT-FIELDNAME = 'LAND1'.  
WA\_FIELDCAT-TABNAME = 'LT\_TAB'.  
APPEND WA\_FIELDCAT TO LT\_FIELDCAT.  
  
ENDFORM.  
  
FORM func\_call .  
  CALL FUNCTION 'REUSE\_ALV\_LIST\_DISPLAY'  
 EXPORTING  
  
    I\_CALLBACK\_PROGRAM             =  SY-REPID  
  
     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.



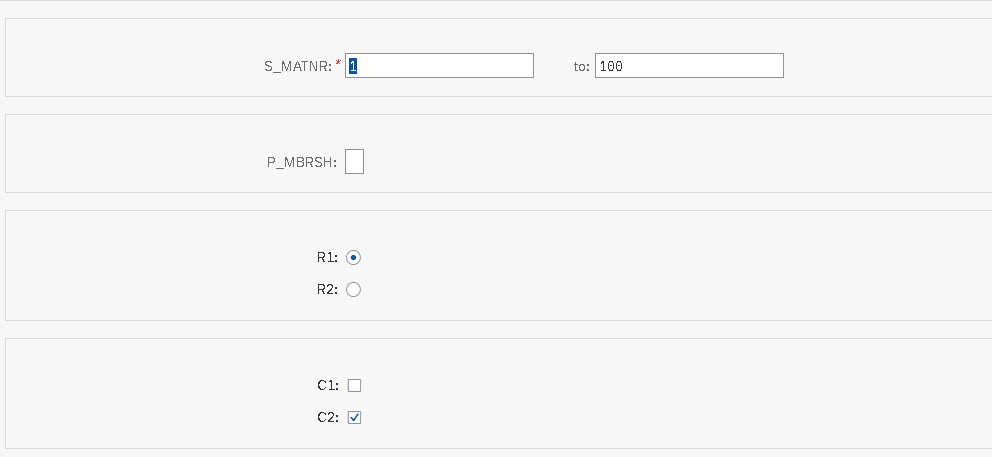
4. ALV GRID

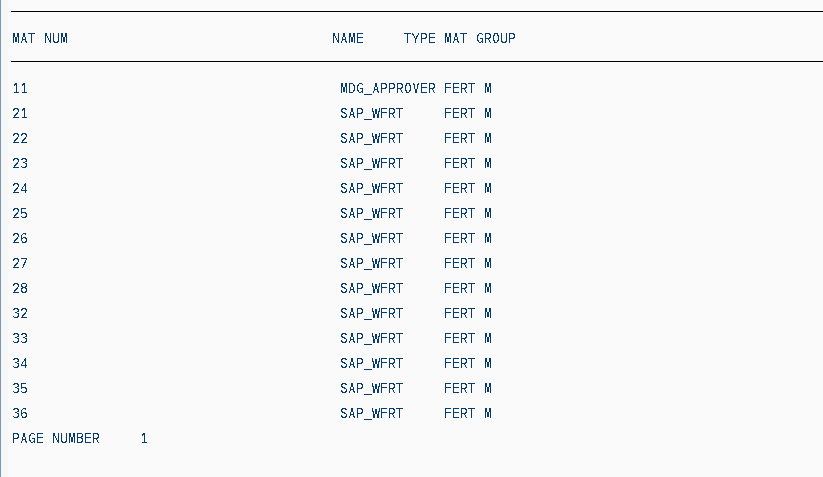
REPORT ZSANG\_ALV\_GRID\_PRO NO STANDARD PAGE HEADING LINE-COUNT 20(3).  
INCLUDE zsang\_alv\_grid\_top.  
SELECT-OPTIONS s\_kunnr for lv\_kunnr.  
INITIALIZATION.  
clear: lt\_fieldcat, lt\_tab.  
s\_kunnr-low = '1000010'.  
s\_kunnr-high = '1000500'.  
append s\_kunnr.  
at SELECTION-SCREEN on s\_kunnr.  
START-OF-SELECTION.  
PERFORM get\_select.  
perform get\_loop.  
end-of-SELECTION.  
perform fun\_call.  
FORM get\_select .  
  
  select kunnr name1 ort01 land1  
    into table lt\_tab  
    from kna1  
    where kunnr in s\_kunnr.  
  
ENDFORM.  
  
FORM get\_loop .  
  wa\_fieldcat-col\_pos = '1'.  
  wa\_fieldcat-fieldname = 'kunnr'.  
  wa\_fieldcat-seltext\_l = 'customer number'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  APPEND wa\_fieldcat to lt\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '2'.  
  wa\_fieldcat-fieldname = 'name1'.  
  wa\_fieldcat-seltext\_l = 'customer name'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  APPEND wa\_fieldcat to lt\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '3'.  
  wa\_fieldcat-fieldname = 'ort01'.  
  wa\_fieldcat-seltext\_l = 'country'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  APPEND wa\_fieldcat to lt\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '4'.  
  wa\_fieldcat-fieldname = 'land1'.  
  wa\_fieldcat-seltext\_l = 'city'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  APPEND wa\_fieldcat to lt\_fieldcat.  
  
ENDFORM.  
  
FORM fun\_call .  
  CALL FUNCTION 'REUSE\_ALV\_GRID\_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\_CALLBACK\_TOP\_OF\_PAGE            = ' '  
\*     I\_CALLBACK\_HTML\_TOP\_OF\_PAGE       = ' '  
\*     I\_CALLBACK\_HTML\_END\_OF\_LIST       = ' '  
\*     I\_STRUCTURE\_NAME                  =  
\*     I\_BACKGROUND\_ID                   = ' '  
\*     I\_GRID\_TITLE                      =  
\*     I\_GRID\_SETTINGS                   =  
\*     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  
\*     I\_HTML\_HEIGHT\_TOP                 = 0  
\*     I\_HTML\_HEIGHT\_END                 = 0  
\*     IT\_ALV\_GRAPHICS                   =  
\*     IT\_HYPERLINK                      =  
\*     IT\_ADD\_FIELDCAT                   =  
\*     IT\_EXCEPT\_QINFO                   =  
\*     IR\_SALV\_FULLSCREEN\_ADAPTER        =  
\*     O\_PREVIOUS\_SRAL\_HANDLER           =  
\*     O\_COMMON\_HUB                      =  
\*   IMPORTING  
\*     E\_EXIT\_CAUSED\_BY\_CALLER           =  
\*     ES\_EXIT\_CAUSED\_BY\_USER            =  
    TABLES  
      t\_outtab                          = LT\_TAB  
  EXCEPTIONS  
       PROGRAM\_ERROR                     = 1  
    OTHERS                            = 2  
            .  
  IF sy-subrc <> 0.  
    "Implement suitable error handling here.  
  ENDIF.  
  
  
ENDFORM.

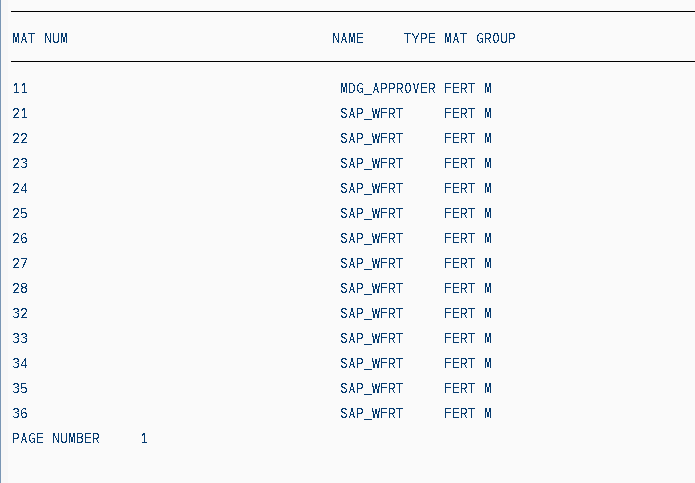


WITH OUTPUT SCREEN.

REPORT zsang\_intwithoutputscreen NO STANDARD PAGE HEADING LINE-COUNT 20(3).  
  
LOAD-OF-PROGRAM."IT WILL INITIALIZE THE PROGRAM INTO THE ABAP MEMORY.  
  INCLUDE zsang\_intout\_top.  
  
  
  
  SELECTION-SCREEN: BEGIN OF BLOCK b1 WITH FRAME TITLE TEXT-001.  
    SELECT-OPTIONS s\_matnr FOR wa\_tab-matnr OBLIGATORY NO-EXTENSION.  
  SELECTION-SCREEN: END OF BLOCK b1.  
  SELECTION-SCREEN: BEGIN OF BLOCK b2 WITH FRAME TITLE TEXT-002.  
    PARAMETERS: p\_mtart TYPE mtart MODIF ID m2,  
                p\_mbrsh TYPE mbrsh MODIF ID m4.  
  SELECTION-SCREEN: END OF BLOCK b2.  
  
  
  SELECTION-SCREEN : BEGIN OF BLOCK b3 WITH FRAME TITLE TEXT-003.  
  
    PARAMETERS : r1 RADIOBUTTON GROUP rad1 USER-COMMAND flag DEFAULT 'X',  
                 r2 RADIOBUTTON GROUP rad1.  
  SELECTION-SCREEN : END OF BLOCK b3.  
  
  SELECTION-SCREEN: BEGIN OF BLOCK b4 WITH FRAME TITLE TEXT-004.  
    PARAMETERS : c1 AS CHECKBOX,  
                 c2 AS CHECKBOX DEFAULT 'X'.  
  SELECTION-SCREEN: END OF BLOCK b4.  
  
  
AT SELECTION-SCREEN OUTPUT.  
  LOOP AT SCREEN.  
    IF r1 = 'X' AND  
      screen-group1 = 'M2'.  
      screen-active = 0.  
    ELSEIF r2 = 'X' AND  
      screen-group1 = 'M4'.  
      screen-active = 0.  
    ENDIF.  
    MODIFY SCREEN.  
  ENDLOOP.  
  
  
INITIALIZATION.  "FOR PROVIDING DEFAULT VALUES, DO CALCUKATIONS, FOR CLEARING.  
  CLEAR: wa\_tab, lt\_tab.  
  s\_matnr-low = '1'.  
  s\_matnr-high = '100'.  
  APPEND  s\_matnr.  
AT SELECTION-SCREEN ON s\_matnr.  
  
  SELECT SINGLE matnr  
    INTO wa\_tab  
  FROM mara  
  WHERE matnr IN s\_matnr.  
  
  IF sy-subrc EQ 0.  
    MESSAGE s000(zsang\_msg).  
  ELSE.  
    MESSAGE s001(zsang\_msg).  
  ENDIF.  
  
TOP-OF-PAGE.  
  ULINE.  
  WRITE: / 'MAT NUM',  
  40 ' NAME',  
  50 'TYPE',  
  55 'MAT GROUP'.  
  ULINE.  
  
END-OF-PAGE.  
  WRITE:/ 'PAGE NUMBER', sy-pagno.  
  
START-OF-SELECTION. "IT IS USED BEFORE EXECUTING THE SQL QUERY. IT PROVIDES BASIC LIST INFO.  
  PERFORM get\_select.  
END-OF-SELECTION.  
  PERFORM get\_loop.  
  
  
FORM get\_select .  
  SELECT matnr ernam mtart mbrsh  
  INTO TABLE lt\_tab  
  FROM mara  
  WHERE matnr IN s\_matnr AND mtart EQ p\_mtart.  
  SORT lt\_tab BY matnr.  
  
  LOOP AT lt\_tab INTO wa\_tab.  
    WRITE : / wa\_tab-matnr,  
    wa\_tab-ernam,  
    wa\_tab-mtart,  
    wa\_tab-mbrsh.  
  ENDLOOP.  
  
ENDFORM.  
  
  
  
FORM get\_loop .  
  SELECT matnr ernam mtart mbrsh  
  INTO TABLE lt\_tab  
  FROM mara  
  WHERE matnr IN s\_matnr AND mbrsh EQ p\_mbrsh.  
  SORT lt\_tab BY matnr.  
  
  LOOP AT lt\_tab INTO wa\_tab.  
    WRITE: / wa\_tab-matnr,  
    wa\_tab-ernam,  
    wa\_tab-mtart,  
    wa\_tab-mbrsh.  
  ENDLOOP.  
  
ENDFORM.







HIERARICHAL

REPORT zsang\_hierarical\_pro.  
INCLUDE zsang\_hierarcal\_top.  
TABLES vbaK.  
SELECT-OPTIONS s\_vbeln FOR vbak-vbeln.  
  
START-OF-SELECTION.  
  PERFORM get\_select.  
  PERFORM create\_fact.  
  
END-OF-SELECTION.  
  PERFORM get\_hierarchical.  
  
  
FORM get\_select .  
  SELECT vbeln erdat ernam vbtyp  
    FROM vbak  
  INTO CORRESPONDING FIELDS OF  TABLE lt\_tab  
  WHERE vbeln IN s\_vbeln.  
  
  SELECT vbeln  
    posnr  
   matnr  
   matkl  
   FROM vbap  
  INTO TABLE lt\_tab1  
  WHERE vbeln IN s\_vbeln.  
  
  
ENDFORM.  
  
FORM create\_fact .  
  
  w\_fcat-col\_pos       = '1'. "coloum position  
  w\_fcat-fieldname     = 'VBELN'. "column name  
  w\_fcat-tabname       = 'LT\_TAB'. "table  
  w\_fcat-ref\_tabname   = 'VBAK'. "table  
  w\_fcat-ref\_fieldname = 'VBELN'. "reference field, it will show descriptions automatically  
  APPEND w\_fcat TO t\_fcat.  
  CLEAR w\_fcat.  
  
  w\_fcat-col\_pos       = '2'.  
  w\_fcat-fieldname     = 'ERDAT'.  
  w\_fcat-tabname       = 'LT\_TAB'.  
  w\_fcat-ref\_tabname   = 'VBAK'.  
  w\_fcat-ref\_fieldname = 'ERDAT'.  
  APPEND w\_fcat TO t\_fcat.  
  CLEAR w\_fcat.  
  
  w\_fcat-col\_pos       = '3'.  
  w\_fcat-fieldname     = 'ERNAM'.  
  w\_fcat-tabname       = 'LT\_TAB'.  
  w\_fcat-ref\_tabname   = 'VBAK'.  
  w\_fcat-ref\_fieldname = 'ERNAM'.  
  APPEND w\_fcat TO t\_fcat.  
  CLEAR w\_fcat.  
  
  w\_fcat-col\_pos       = '4'.  
  w\_fcat-fieldname     = 'VBTYP'.  
  w\_fcat-tabname       = 'LT\_TAB'.  
  w\_fcat-ref\_tabname   = 'VBAK'.  
  w\_fcat-ref\_fieldname = 'VBTYP'.  
  APPEND w\_fcat TO t\_fcat.  
  CLEAR w\_fcat.  
  
  
  
\*\*\*build fcat for VBAP  
  w\_fcat-col\_pos       = '1'. "coloum position  
  w\_fcat-fieldname     = 'VBELN'. "column name  
  w\_fcat-tabname       = 'LT\_TAB1'. "table  
  w\_fcat-ref\_tabname   = 'VBAP'. "table  
  w\_fcat-ref\_fieldname = 'VBELN'. "reference field, it will show descriptions automatically  
  APPEND w\_fcat TO t\_fcat.  
  CLEAR w\_fcat.  
  
  w\_fcat-col\_pos       = '2'. "coloum position  
  w\_fcat-fieldname     = 'POSNR'. "column name  
  w\_fcat-tabname       = 'LT\_TAB1'. "table  
  w\_fcat-ref\_tabname   = 'VBAP'. "table  
  w\_fcat-ref\_fieldname = 'POSNR'. "reference field, it will show descriptions automatically  
  APPEND w\_fcat TO t\_fcat.  
  CLEAR w\_fcat.  
  
  w\_fcat-col\_pos       = '3'. "coloum position  
  w\_fcat-fieldname     = 'MATNR'. "column name  
  w\_fcat-tabname       = 'LT\_TAB1'. "table  
  w\_fcat-ref\_tabname   = 'VBAP'. "table  
  w\_fcat-ref\_fieldname = 'MATNR'. "reference field, it will show descriptions automatically  
  APPEND w\_fcat TO t\_fcat.  
  CLEAR w\_fcat.  
  w\_fcat-col\_pos       = '4'. "coloum position  
  w\_fcat-fieldname     = 'MATKL'. "column name  
  w\_fcat-tabname       = 'LT\_TAB1'. "table  
  w\_fcat-ref\_tabname   = 'VBAP'. "table  
  w\_fcat-ref\_fieldname = 'MATKL'. "reference field, it will show descriptions automatically  
  APPEND w\_fcat TO t\_fcat.  
  CLEAR w\_fcat.  
  
  
ENDFORM.  
  
FORM get\_hierarchical .  
  it\_key-header01 =  'VBELN'.  
  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          = layout  
        it\_fieldcat        = t\_fcat[]  
\*       IT\_EXCLUDING       =  
\*       IT\_SPECIAL\_GROUPS  =  
\*       IT\_SORT            =  
\*       IT\_FILTER          =  
\*       IS\_SEL\_HIDE        =  
\*       I\_SCREEN\_START\_COLUMN          = 0  
\*       I\_SCREEN\_START\_LINE            = 0  
\*       I\_SCREEN\_END\_COLUMN            = 0  
\*       I\_SCREEN\_END\_LINE  = 0  
\*       I\_DEFAULT          = 'X'  
\*       I\_SAVE             = ' '  
\*       IS\_VARIANT         =  
\*       IT\_EVENTS          =  
\*       IT\_EVENT\_EXIT      =  
        i\_tabname\_header   = 'LT\_TAB'  
        i\_tabname\_item     = 'LT\_TAB1'  
\*       I\_STRUCTURE\_NAME\_HEADER        =  
\*       I\_STRUCTURE\_NAME\_ITEM          =  
        is\_keyinfo         = it\_key  
\*       IS\_PRINT           =  
\*       IS\_REPREP\_ID       =  
\*       IS\_HIERSEQ\_SETTINGS            =  
\*       I\_BYPASSING\_BUFFER =  
\*       I\_BUFFER\_ACTIVE    =  
\*       IR\_SALV\_HIERSEQ\_ADAPTER        =  
\*       IT\_EXCEPT\_QINFO    =  
\*       I\_SUPPRESS\_EMPTY\_DATA          = ABAP\_FALSE  
\*       I\_LIST\_DISPLAY\_ONLY            = ABAP\_FALSE  
\*       O\_COMMON\_HUB       =  
\*     IMPORTING  
\*       E\_EXIT\_CAUSED\_BY\_CALLER        =  
\*       ES\_EXIT\_CAUSED\_BY\_USER         =  
      TABLES  
        t\_outtab\_header    = lt\_tab  
        t\_outtab\_item      = lt\_tab1  
      EXCEPTIONS  
        program\_error      = 1  
        OTHERS             = 2.  
    IF sy-subrc <> 0.  
\* Implement suitable error handling here  
    ENDIF.  
  else.  
  WRITE 'No records found'.  
  endif.  
  
  
ENDFORM.

