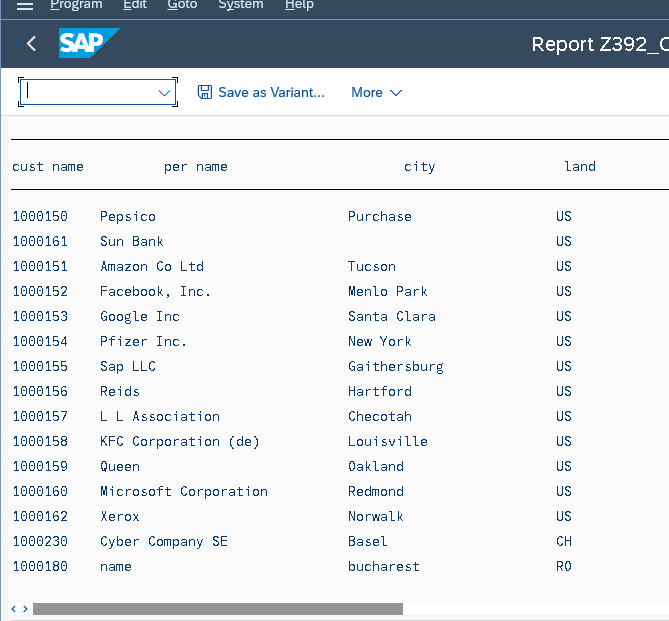
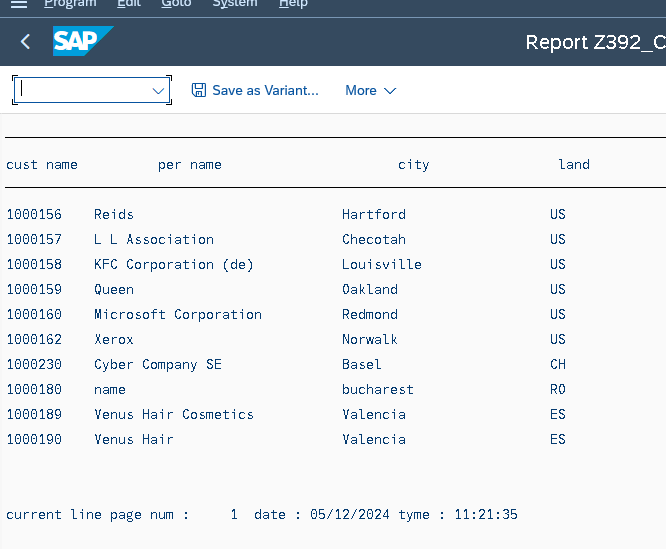
WEEK 2

🡺classical

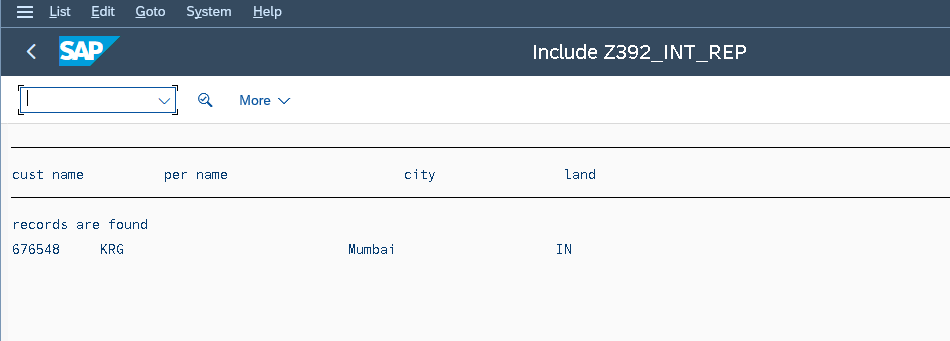
REPORT Z392\_CLS\_REP no STANDARD PAGE HEADING LINE-COUNT 23(3) LINE-SIZE 500.  
  
load-OF-PROGRAM.  
include zdharani\_01.  
  
select-OPTIONS s\_kunnr for wa\_tab-kunnr OBLIGATORY.  
  
INITIALIZATION.  
clear: it\_tab, wa\_tab.  
s\_kunnr-low = '1000150'.  
s\_kunnr-high = '1000500'.  
append s\_kunnr.  
  
at SELECTION-SCREEN on s\_kunnr.  
  select single kunnr into lv\_kunnr  
    from kna1  
    where kunnr in s\_kunnr.  
  
START-OF-SELECTION.  
set PF-STATUS 'ZDHAR\_PFS'.  
  perform get\_kna1\_data.  
  
end-of-SELECTION.  
  perform dis\_kna1\_data.  
  
  
  TOP-OF-PAGE.  
  uline.  
  write :/ 'cust name',  
          20 'per name',  
          50 'city',  
          70  'land'.  
  ULINE.  
  
  
 end-of-PAGE.  
 skip 2.  
  write :/ 'current line page num :', sy-pagno,  
           'date :', sy-datum,  
           'tyme :', sy-uzeit.  
  
  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.  
  
  at USER-COMMAND.  
    case sy-ucomm.  
      WHEN 'DISP'.  
        WRITE 'UST INFO'.  
       when 'SAP'.  
    write 'SAP LABS BNGL '.  
  
when 'ABAP'.  
    write 'SAP ABAP HYD '.  
  
  endcase.  
  
  
\*&---------------------------------------------------------------------\*  
\*& Form get\_kna1\_data  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM get\_kna1\_data .  
  
  select kunnr name1 ort01 land1 into table it\_tab  
    from kna1  
    where kunnr in s\_kunnr.  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form dis\_kna1\_data  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM dis\_kna1\_data .  
  
  loop at it\_tab into wa\_tab.  
    write :/ wa\_tab-kunnr,  
             wa\_tab-name1,  
             wa\_tab-ort01,  
             wa\_tab-land1.  
  ENDLOOP.  
ENDFORM.





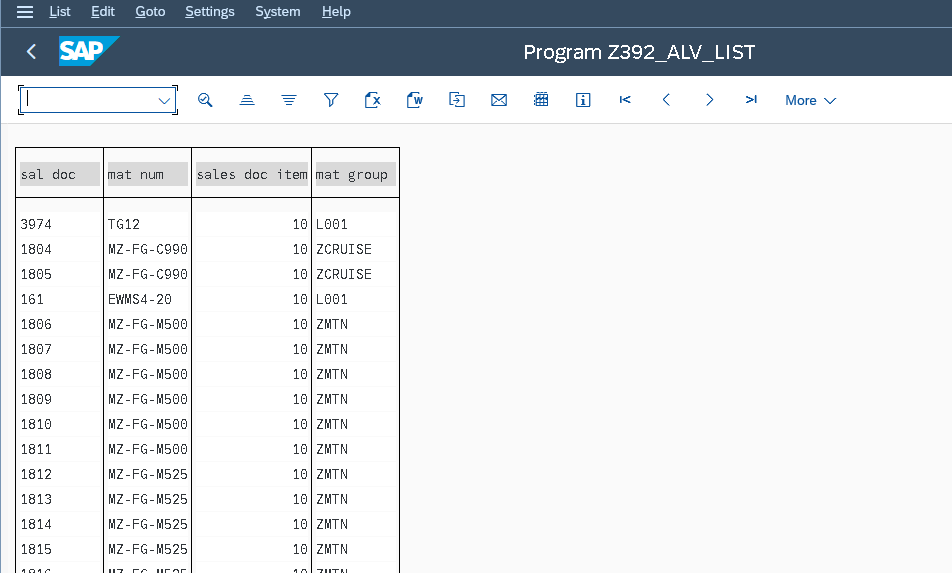
🡺Interactive

REPORT Z392\_INT\_REP no STANDARD PAGE HEADING LINE-COUNT 23(3) LINE-SIZE 500.  
  
load-OF-PROGRAM.  
include zdharani\_01.  
  
select-OPTIONS s\_kunnr for wa\_tab-kunnr OBLIGATORY.  
  
INITIALIZATION.  
CLEAR : IT\_TAB,WA\_TAB.  
s\_kunnr-low = '100'.  
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.  
  
  
  
START-OF-SELECTION.  
\*  set PF-STATUS 'ZDHAR\_PFS'.  
  perform get\_kna1\_data.  
  
end-of-SELECTION.  
  perform dis\_kna1\_data.  
  
  
  TOP-OF-PAGE.  
  uline.  
  write :/ 'cust name',  
          20 'per name',  
          50 'city',  
          70  'land'.  
  ULINE.  
  
  
 end-of-PAGE.  
 SKIP 2.  
  write :/ 'current line page num :', sy-pagno,  
           'date :', sy-datum,  
           'tyme :', sy-uzeit.  
  
  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.  
  
\*  at USER-COMMAND.  
\*    case sy-ucomm.  
\*      WHEN 'DISP'.  
\*        WRITE 'UST INFO'.  
\*      WRITE :/ 'THE CURRENTV DATE IS', SY-DATUM,  
\*               'TIME IS', SY-UZEIT.  
\*  
\*       when 'SAP'.  
\*    write 'SAP LABS BNGL '.  
\*  
\*when 'ABAP'.  
\*    write 'SAP ABAP HYD '.  
\*  
\*  endcase.  
  
  
  
\*&---------------------------------------------------------------------\*  
\*& Form get\_kna1\_data  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM get\_kna1\_data .  
  
  select kunnr name1 ort01 land1 into table it\_tab  
    from kna1  
    where kunnr in s\_kunnr.  
  
if sy-subrc eq 0.  
  write :/ 'records are found'.  
else.  
  write :/ 'records are not found'.  
  
ENDIF.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form dis\_kna1\_data  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM dis\_kna1\_data .  
  
  loop at it\_tab into wa\_tab.  
    write :/ wa\_tab-kunnr,  
             wa\_tab-name1,  
             wa\_tab-ort01,  
             wa\_tab-land1.  
  ENDLOOP.  
ENDFORM.



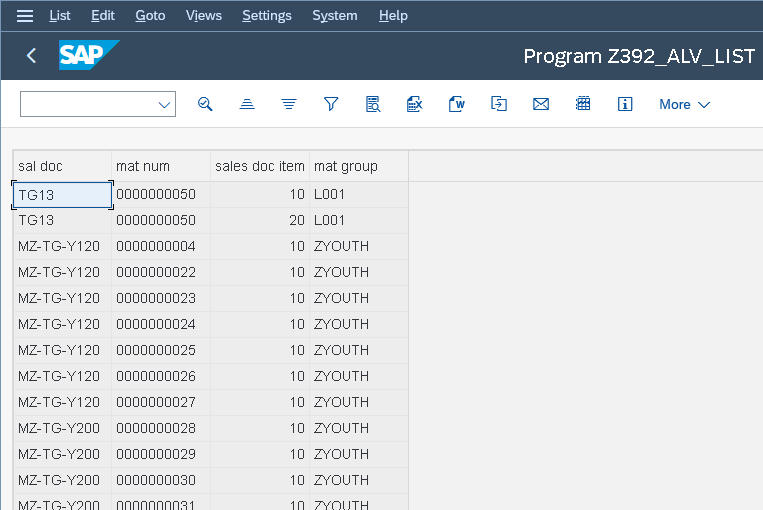
🡺ALV\_LIST

REPORT Z392\_ALV\_LIST.  
include Z392\_ALV\_LIST\_top.  
  
SELECT-OPTIONS S\_vbeln FOR LV\_vbeln OBLIGATORY.  
  
INITIALIZATION.  
clear : it\_tab, wa\_fieldcat, lt\_fieldcat.  
S\_vbeln-LOW = '1'.  
S\_vbeln-HIGH = '10000'.  
APPEND s\_vbeln.  
  
AT SELECTION-SCREEN ON S\_vbeln.  
  SELECT SINGLE vbeln  
    INTO LV\_vbeln  
    FROM vbap  
    WHERE vbeln IN S\_vbeln.  
  
START-OF-SELECTION.  
    PERFORM GET\_ALV\_FCAT.  
  
    perform get\_mat\_data.  
  
end-of-SELECTION.  
  
PERFORM display\_alvout.  
  
  
FORM get\_mat\_data .  
  SELECT vbeln matnr posnr matkl  
    INTO TABLE it\_tab  
    FROM vbap  
    WHERE vbeln IN s\_vbeln.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALV\_FCAT  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM get\_alv\_fcat .  
  
  wa\_fieldcat-col\_pos = '1'.  
  wa\_fieldcat-seltext\_l = 'sal doc'.  
  wa\_fieldcat-fieldname = 'vbeln'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '2'.  
  wa\_fieldcat-seltext\_l = 'mat num'.  
  wa\_fieldcat-fieldname = 'matnr'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '3'.  
  wa\_fieldcat-seltext\_l = 'sales doc item'.  
  wa\_fieldcat-fieldname = 'posnr'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
   append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '4'.  
  wa\_fieldcat-seltext\_l = 'mat group'.  
  wa\_fieldcat-fieldname = 'matkl'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form get\_mat\_data  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
\*FORM get\_mat\_data .  
\*  select matnr  vbeln posnr matkl  
\*    into table it\_tab  
\*    from vbap  
\*    where vbeln in s\_vbeln.  
\*  
\*ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form display\_alvout  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM display\_alvout .  
  CALL FUNCTION 'REUSE\_ALV\_LIST\_DISPLAY'  
 EXPORTING  
\*     I\_INTERFACE\_CHECK              = ' '  
\*     I\_BYPASSING\_BUFFER             =  
\*     I\_BUFFER\_ACTIVE                = ' '  
     I\_CALLBACK\_PROGRAM             = SY-REPID                      
     IT\_FIELDCAT                    =  lt\_fieldcat  
    TABLES  
      t\_outtab                       = it\_tab  
  EXCEPTIONS  
   PROGRAM\_ERROR                  = 1  
     OTHERS                         = 2  
            .  
  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
  ENDIF.  
  
  
ENDFORM.



🡺ALV\_GRID

REPORT Z392\_ALV\_GRIDD.  
include Z392\_ALV\_LIST\_top.  
  
SELECT-OPTIONS S\_vbeln FOR LV\_vbeln OBLIGATORY.  
  
INITIALIZATION.  
clear : it\_tab, wa\_fieldcat, lt\_fieldcat.  
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.  
  
START-OF-SELECTION.  
    PERFORM GET\_ALV\_FCAT.  
  
    perform get\_mat\_data.  
  
end-of-SELECTION.  
  
PERFORM display\_alvout.  
  
FORM get\_alv\_fcat .  
  
  wa\_fieldcat-col\_pos = '1'.  
  wa\_fieldcat-seltext\_l = 'sal doc'.  
  wa\_fieldcat-fieldname = 'vbeln'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '2'.  
  wa\_fieldcat-seltext\_l = 'mat num'.  
  wa\_fieldcat-fieldname = 'matnr'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
  append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '3'.  
  wa\_fieldcat-seltext\_l = 'sales doc item'.  
  wa\_fieldcat-fieldname = 'posnr'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
   append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '4'.  
  wa\_fieldcat-seltext\_l = 'mat group'.  
  wa\_fieldcat-fieldname = 'matkl'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
append wa\_fieldcat to lt\_fieldcat.  
  clear wa\_fieldcat.  
  
ENDFORM.  
  
FORM get\_mat\_data .  
  select matnr  vbeln posnr matkl  
    into table it\_tab  
    from vbap  
    where vbeln in s\_vbeln.  
  
ENDFORM.  
  
FORM display\_alvout .  
  CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
 EXPORTING  
  
     I\_CALLBACK\_PROGRAM             = 'Z392\_ALV\_LIST '  
     IT\_FIELDCAT                    =  lt\_fieldcat  
  
    TABLES  
      t\_outtab                       = it\_tab  
  EXCEPTIONS  
   PROGRAM\_ERROR                  = 1  
     OTHERS                         = 2  
            .  
  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
  ENDIF.  
  
  
ENDFORM.



🡺WITH OUTPUT SCREEN

Report z392\_at\_ssop NO STANDARD PAGE HEADING LINE-COUNT 20(3) LINE-SIZE 500.  
  
load-OF-PROGRAM.  
  
  INCLUDE z392\_at\_ssop1.  
  
SELECTION-SCREEN : BEGIN OF BLOCK B1 WITH FRAME TITLE TEXT-T01.  
  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-T02.  
    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-T03.  
    PARAMETERS : RAD1 RADIOBUTTON GROUP RAD USER-COMMAND FLAG DEFAULT 'X',  
                RAD2 RADIOBUTTON GROUP RAD.  
    SELECTION-SCREEN END OF BLOCK B3.  
  
  
  SELECTION-SCREEN: BEGIN OF BLOCK b4 WITH FRAME TITLE TEXT-T04.  
    PARAMETERS : c1 AS CHECKBOX,  
                 c2 AS CHECKBOX DEFAULT 'X'.  
  SELECTION-SCREEN: END OF BLOCK b4.  
  
  
at SELECTION-SCREEN OUTPUT.  
  loop AT SCREEN.  
    if rad1 = 'x' and  
      screen-group1 = 'M2'.  
      SCREEN-ACTIVE = 0.  
  
    ELSEIF RAD2 = 'X' AND  
    SCREEN-GROUP1 = 'M4'.  
    SCREEN-ACTIVE = 0.  
    ENDIF.  
    MODIFY SCREEN.  
    ENDLOOP.  
  
  
INITIALIZATION.  
clear: wa\_tab, it\_tab.  
  s\_matnr-low = '1'.  
  s\_matnr-high = '100'.  
  APPEND s\_matnr.  
  
AT SELECTION-SCREEN ON s\_matnr.  
  
  SELECT SINGLE matnr  
    INTO s\_matnr  
    FROM mara  
    WHERE matnr IN s\_matnr.  
  
  
IF RAD1 = 'X'.  
  PERFORM get\_data.  
  
ELSE.  
  PERFORM get\_display.  
 ENDIF.  
  
TOP-OF-PAGE.  
  ULINE.  
  WRITE :/ 'MAT NUM',  
        30 'NAME',  
        50 'TYPE',  
        70 'MAT GRUP'.  
  ULINE.  
  
END-OF-PAGE.  
  WRITE :/ 'current list page no:' , sy-pagno,  
            'THE CURRENT DATE', sy-datum,  
           'TYPE',sy-uzeit.  
  
  
*\*&---------------------------------------------------------------------\**  
*\*& Form GET\_DATA*  
*\*&---------------------------------------------------------------------\**  
*\*& text*  
*\*&---------------------------------------------------------------------\**  
*\*& -->  p1        text*  
*\*& <--  p2        text*  
*\*&---------------------------------------------------------------------\**  
FORM get\_data .  
  
  
  SELECT matnr ernam mtart mbrsh  
       INTO TABLE it\_tab  
       FROM mara  
  WHERE matnr IN s\_matnr AND MBRSH EQ P\_MBRSH.  
  
    LOOP AT It\_tab INTO wa\_tab.  
    WRITE : / wa\_tab-matnr,  
              wa\_tab-ernam,  
              wa\_tab-mtart,  
             wa\_tab-mbrsh.  
  ENDLOOP.  
  
  IF sy-subrc EQ 0.  
    WRITE :/ 'Records are found in mara table'.  
  ELSE.  
    WRITE : / 'Records are not found in mara table'.  
  ENDIF.  
  
ENDFORM.  
*\*&---------------------------------------------------------------------\**  
*\*& Form GET\_DISPLAY*  
*\*&---------------------------------------------------------------------\**  
*\*& text*  
*\*&---------------------------------------------------------------------\**  
*\*& -->  p1        text*  
*\*& <--  p2        text*  
*\*&---------------------------------------------------------------------\**  
FORM get\_display .  
  SELECT matnr ernam mtart mbrsh  
       INTO TABLE it\_tab  
       FROM mara  
  WHERE matnr IN s\_matnr AND MTART EQ P\_MTART.  
  
  LOOP AT It\_tab INTO wa\_tab.  
    WRITE : / wa\_tab-matnr,  
             30 wa\_tab-ernam,  
             40 wa\_tab-mtart,  
             55 wa\_tab-mbrsh.  
  ENDLOOP.  
  
ENDFORM.

A screenshot of a computer

Description automatically generated

🡺HIERARCHICAL

REPORT Z392\_HIER.  
types : begin of ls\_vbak,  
  UST(1),  
   vbeln type vbeln\_VA,  
   erdat type erdat,  
  erzet type erzet,  
  END OF ls\_vbak.  
  
data : it\_vbak type table of ls\_vbak,  
       wa\_tab type ls\_vbak.  
  
types : begin of ls\_vbap,  
       vbeln type vbeln\_va,  
       posnr type posnr\_va,  
      matnr type matnr,  
  END OF ls\_vbap.  
  
data : it\_vbap type table of ls\_vbap,  
       wa\_vbap type ls\_vbap,  
  
 it\_fcat TYPE slis\_t\_fieldcat\_alv,  
  wa\_fcat LIKE LINE OF it\_fcat,  
  
it\_key TYPE slis\_keyinfo\_alv,  
  
 layout TYPE slis\_layout\_alv.  
  
tables vbak.  
  select-OPTIONS s\_vbeln for vbak-vbeln.  
  
  START-OF-SELECTION.  
  perform get\_data.  
  
  PERFORM create\_col.  
  
  PERFORM hierdisplay.  
  
  
FORM get\_data .  
  select vbeln erdat erzet  
    from vbak  
    into CORRESPONDING FIELDS OF table it\_vbak  
    where vbeln in s\_vbeln.  
  
    select vbeln posnr matnr  
      from vbap  
      into table it\_vbap  
      where vbeln in s\_vbeln.  
  
  
ENDFORM.  
  
FORM create\_col .  
   wa\_fcat-col\_pos = '1'.  
   wa\_fcat-tabname = 'IT\_VBAK'.  
   wa\_fcat-fieldname = 'VBELN'.  
   wa\_fcat-ref\_tabname = 'VBAK'.  
   wa\_fcat-ref\_fieldname = 'VBELN'.  
  
 append wa\_fcat to it\_fcat.  
   clear wa\_fcat.  
  
   wa\_fcat-col\_pos = '2'.  
   wa\_fcat-tabname = 'IT\_VBAK'.  
   wa\_fcat-fieldname = 'ERDAT'.  
   wa\_fcat-ref\_tabname = 'VBAK'.  
   wa\_fcat-ref\_fieldname = 'ERDAT'.  
  
 append wa\_fcat to it\_fcat.  
   clear wa\_fcat.  
  
   wa\_fcat-col\_pos = '3'.  
   wa\_fcat-tabname = 'IT\_VBAK'.  
   wa\_fcat-fieldname = 'ERZET'.  
   wa\_fcat-ref\_tabname = 'VBAK'.  
   wa\_fcat-ref\_fieldname = 'ERZET'.  
  
 append wa\_fcat to it\_fcat.  
   clear wa\_fcat.  
  
   wa\_fcat-col\_pos = '1'.  
   wa\_fcat-tabname = 'IT\_VBAP'.  
   wa\_fcat-fieldname = 'VBELN'.  
   wa\_fcat-ref\_tabname = 'VBAP'.  
   wa\_fcat-ref\_fieldname = 'VBELN'.  
  
 append wa\_fcat to it\_fcat.  
   clear wa\_fcat.  
  
   wa\_fcat-col\_pos = '2'.  
   wa\_fcat-tabname = 'IT\_VBAP'.  
   wa\_fcat-fieldname = 'POSNR'.  
   wa\_fcat-ref\_tabname = 'VBAP'.  
   wa\_fcat-ref\_fieldname = 'POSNR'.  
  
 append wa\_fcat to it\_fcat.  
   clear wa\_fcat.  
  
   wa\_fcat-col\_pos = '3'.  
   wa\_fcat-tabname = 'IT\_VBAP'.  
   wa\_fcat-fieldname = 'MATNR'.  
   wa\_fcat-ref\_tabname = 'VBAP'.  
   wa\_fcat-ref\_fieldname = 'MATNR'.  
  
 append wa\_fcat to it\_fcat.  
   clear wa\_fcat.  
  
  
  
ENDFORM.  
*\*&---------------------------------------------------------------------\**  
*\*& Form hierdisplay*  
*\*&---------------------------------------------------------------------\**  
*\*& text*  
*\*&---------------------------------------------------------------------\**  
*\*& -->  p1        text*  
*\*& <--  p2        text*  
*\*&---------------------------------------------------------------------\**  
FORM hierdisplay .  
  it\_key-header01 =  'VBELN'.  
  layout-expand\_fieldname = 'UST'.  
  
  
  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                    = it\_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               = 'IT\_VBAK'  
                    i\_tabname\_item                 =  'IT\_VBAP'  
*\*                   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                = IT\_VBAK  
                    t\_outtab\_item                  = IT\_VBAP  
                EXCEPTIONS  
                   PROGRAM\_ERROR                  = 1  
                   OTHERS                         = 2  
                          .  
                IF sy-subrc <> 0.  
*\* Implement suitable error handling here*  
                ENDIF.  
  
ENDIF.  
  
  
ENDFORM.

A screenshot of a software program

Description automatically generated

## 🡺GET CURSOR

data lv\_vbeln type vbeln\_va.  
  
types: begin of ls\_vbak,  
  vbeln type vbeln\_va,  
  kunnr type kunnr,  
  erdat type erdat,  
  END OF ls\_vbak.  
  
data : it\_vbak type TABLE of ls\_vbak,  
       wa\_vbak type ls\_vbak.  
  
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 :/ wa\_vbak-vbeln,  
              wa\_vbak-kunnr.  
      FORMAT HOTSPOT off.  
      write :/ wa\_vbak-erdat.  
      ENDLOOP.  
  
TOP-OF-PAGE.  
      write :/ 'sales doc num',  
               'customer num',  
               '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'.   *" kunnr 123*  
  SET PARAMETER ID   'KUN'   FIELD fval.  
  CALL TRANSACTION 'XD02' AND SKIP FIRST SCREEN.  
ENDIF.

A screenshot of a computer

Description automatically generated

=> ALV\_LOGO

REPORT z392\_alv\_logo NO STANDARD PAGE HEADING LINE-COUNT 20(3).  
INCLUDE Z392\_ALV\_LIST\_top.  
  
SELECT-OPTIONS S\_vbeln FOR LV\_vbeln OBLIGATORY.  
  
INITIALIZATION.  
  CLEAR : it\_tab, wa\_fieldcat, lt\_fieldcat.  
  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.  
  
  
START-OF-SELECTION.  
  PERFORM get\_alv\_fcat.  
  
  PERFORM get\_mat\_data.  
  
END-OF-SELECTION.  
 PERFORM display\_alvout.  
  
\*&---------------------------------------------------------------------\*  
\*& Form GET\_ALV\_FCAT  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM get\_alv\_fcat .  
  
  wa\_fieldcat-col\_pos = '1'.  
  wa\_fieldcat-seltext\_l = text-001.  
  wa\_fieldcat-fieldname = cs\_vbeln.  
  wa\_fieldcat-tabname = cs\_tab.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '2'.  
  wa\_fieldcat-seltext\_l = text-002.  
  wa\_fieldcat-fieldname = cs\_matnr.  
  wa\_fieldcat-tabname = cs\_tab.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '3'.  
  wa\_fieldcat-seltext\_l = text-003.  
  wa\_fieldcat-fieldname = 'posnr'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
  wa\_fieldcat-col\_pos = '4'.  
  wa\_fieldcat-seltext\_l = text-004.  
  wa\_fieldcat-fieldname = 'matkl'.  
  wa\_fieldcat-tabname = 'lt\_tab'.  
  
  APPEND wa\_fieldcat TO lt\_fieldcat.  
  CLEAR wa\_fieldcat.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form get\_mat\_data  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM get\_mat\_data .  
  SELECT   vbeln MATNR posnr matkl  
    INTO TABLE it\_tab  
    FROM vbap  
    WHERE vbeln IN s\_vbeln.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form display\_alvout  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM display\_alvout .  
  
  data lv\_title type lvc\_title value ' MAT MASTER INFO'.  
  
  ty\_layout-no\_colhead = 'X'.  
  ty\_layout-no\_vline = 'X' .  
  ty\_layout-no\_hline = 'X'.  
  
  if sy-subrc eq 0.  
  
  
  
CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
  EXPORTING  
    I\_CALLBACK\_PROGRAM                =  SY-REPID  
\*     I\_CALLBACK\_PF\_STATUS\_SET          = ' '  
\*     I\_CALLBACK\_USER\_COMMAND           = ' '  
    I\_CALLBACK\_TOP\_OF\_PAGE            =  'UST\_LOGO'  
    I\_GRID\_TITLE                      =  lv\_title  
\*     I\_GRID\_SETTINGS                   =  
\*     IS\_LAYOUT                         =  
     IT\_FIELDCAT                       = LT\_FIELDCAT  
  
    TABLES  
      t\_outtab                          = IT\_TAB  
 EXCEPTIONS  
    PROGRAM\_ERROR                     = 1  
    OTHERS                            = 2  
            .  
  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
  ENDIF.  
ENDIF.  
  
ENDFORM.  
  
  
  
FORM UST\_LOGO.  
  DATA : lt\_header TYPE slis\_t\_listheader,  
         wa\_header TYPE slis\_listheader.  
  
  wa\_header-typ = 's'.  
    wa\_header-info = 'UST GLOBAL'.  
  wa\_header-key = 'HYDERABAD'.  
  append wa\_header to lt\_header.  
  
  wa\_header-typ = 'S'.  
  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                   = 'Z292\_LOGO'  
\*   I\_END\_OF\_LIST\_GRID       =  
\*   I\_ALV\_FORM               =  
          .  
  
ENDFORM.

A screenshot of a computer

Description automatically generated

=> FOR ALL ENTRIES

REPORT Z392\_FORALL.  
  
include Z392\_FORALL\_TOP.  
  
TABLES mara.  
  
select-OPTIONS s\_matnr for mara-matnr.  
  
START-OF-SELECTION.  
select matnr  
       ernam  
       mtart  
       matkl  
  into table it\_tab1  
  from mara  
  where matnr in s\_matnr.  
  
if not it\_tab1[] is INITIAL . " initial  
  select matnr  
         werks  
        pstat  
    into table it\_tab2  
    from marc  
  
FOR ALL ENTRIES IN it\_tab1 "2  
    where matnr eq it\_tab1-matnr.  
  
delete ADJACENT DUPLICATES FROM it\_tab1 COMPARING matnr. "3  
    sort it\_tab2 by matnr.  
  
    loop at it\_tab1 into wa\_tab1.  
  
READ TABLE it\_tab2 into wa\_tab2 with key matnr = wa\_tab1-matnr BINARY SEARCH.  
  
      if sy-subrc = 0.  
  
move : wa\_tab1-matnr to wa\_final-matnr,  
       wa\_tab1-ernam to wa\_final-ernam,  
       wa\_tab1-mtart to wa\_final-mtart,  
       wa\_tab1-matkl to wa\_final-matkl,  
  
       wa\_tab2-werks to wa\_final-werks,  
       wa\_tab2-pstat to wa\_final-pstat.  
  
append wa\_final to it\_final.  
  
else.  
  CONTINUE.  
  ENDIF.  
  ENDLOOP.  
  
loop at it\_final into wa\_final.  
  write :/ wa\_final-matnr,  
           wa\_final-ernam,  
           wa\_final-mtart,  
           wa\_final-matkl,  
           wa\_final-werks,  
           wa\_final-pstat.  
  
  
  
endloop.  
  
else.  
  
  write:/ 'records are not found'.  
  
endif.

A screenshot of a computer program

Description automatically generated

=> FOR ALL ENTRIES with 3 tab.

REPORT Z392\_FORALL\_3TAB.  
  
INCLUDE z392\_forall\_3tab\_top.  
  
select-OPTIONS s\_matnr for wa\_mara-matnr.  
  
INITIALIZATION.  
s\_matnr-low = '1'.  
s\_matnr-high = '100'.  
append s\_matnr.  
  
START-OF-SELECTION.  
  
select matnr ersda ernam mtart  
  into table it\_mara  
  from mara  
  where matnr in s\_matnr.  
if it\_mara[] is not initial. *" checks weather it\_mara has values or not*  
*\*  write ' entries found '.*  
  select matnr werks pstat into table it\_marc  
    from marc                     *" it scans throu8ght all the entries in it\_mara and checks entry aganist conition*  
    for ALL ENTRIES in it\_mara  
    where matnr = it\_mara-matnr.  
  
 if it\_marc[] is not initial.  
   select matnr maktx  
     into table it\_makt  
     from makt  
     for ALL ENTRIES IN it\_marc  
     where matnr = it\_marc-matnr.  
 endif.  
  
 delete ADJACENT DUPLICATES FROM it\_mara COMPARING matnr.  
  
sort it\_mara BY matnr.  
  
  
  
  loop at it\_mara into wa\_mara.  
    read table it\_marc into wa\_marc WITH KEY matnr = wa\_mara-matnr BINARY SEARCH.  
      READ TABLE it\_makt into wa\_makt WITH KEY matnr = wa\_marc-matnr BINARY SEARCH.  
  
      MOVE: wa\_mara-matnr to wa\_final-matnr,  
      wa\_mara-ersda to wa\_final-ersda,  
      wa\_mara-ernam to wa\_final-ernam,  
      wa\_mara-mtart to wa\_final-mtart,  
      wa\_marc-werks to wa\_final-werks,  
      wa\_marc-PSTAT to wa\_final-PSTAT,  
      wa\_makt-maktx to wa\_final-maktx.  
  
APPEND wa\_final to it\_final.  
ENDLOOP.  
  
  
 perform display\_data.  
  
else.  
  write 'no entries '.  
endif.  
*\*&---------------------------------------------------------------------\**  
*\*& Form display\_data*  
*\*&---------------------------------------------------------------------\**  
*\*& text*  
*\*&---------------------------------------------------------------------\**  
*\*& -->  p1        text*  
*\*& <--  p2        text*  
*\*&---------------------------------------------------------------------\**  
FORM display\_data .  
  
cl\_demo\_output=>display( it\_final ).  
  
  
  
ENDFORM.

A screenshot of a software

Description automatically generated

=> SUBROUTINES

REPORT Z392\_SUBROUTINES.  
  
\*data a type i value 10.  
\*perform sub using a.  
\*write a.  
\*\*&---------------------------------------------------------------------\*  
\*\*& Form sub  
\*\*&---------------------------------------------------------------------\*  
\*\*& text  
\*\*&---------------------------------------------------------------------\*  
\*\*&      --> A  
\*\*&---------------------------------------------------------------------\*  
\*FORM sub USING p\_a.  
\*  p\_a = 50.  
\*  
\*  
\*ENDFORM.  
  
  
\*data a type i value 10.  
\*perform sub changing a.  
\*write a.  
\*\*&---------------------------------------------------------------------\*  
\*\*& Form sub  
\*\*&---------------------------------------------------------------------\*  
\*\*& text  
\*\*&---------------------------------------------------------------------\*  
\*\*&      <-- A  
\*\*&---------------------------------------------------------------------\*  
\*FORM sub  CHANGING p\_a.  
\*  p\_a = 20.  
\*  
\*ENDFORM.  
  
\*data c type i value 10.  
\*perform sub using c.  
\*  
\*write c.  
\*\*&---------------------------------------------------------------------\*  
\*\*& Form sub  
\*\*&---------------------------------------------------------------------\*  
\*\*& text  
\*\*&---------------------------------------------------------------------\*  
\*\*&      --> C  
\*\*&---------------------------------------------------------------------\*  
\*FORM sub  USING    p\_c.  
\*  p\_c = 20.  
\*  
\*ENDFORM.  
  
\*data a type i value 10.  
\*perform sub changing a.  
\*write a.  
\*\*&---------------------------------------------------------------------\*  
\*\*& Form sub  
\*\*&---------------------------------------------------------------------\*  
\*\*& text  
\*\*&---------------------------------------------------------------------\*  
\*\*&      <-- A  
\*\*&---------------------------------------------------------------------\*  
\*FORM sub  CHANGING p\_a.  
\*  p\_a = 50.  
\*  
\*ENDFORM.