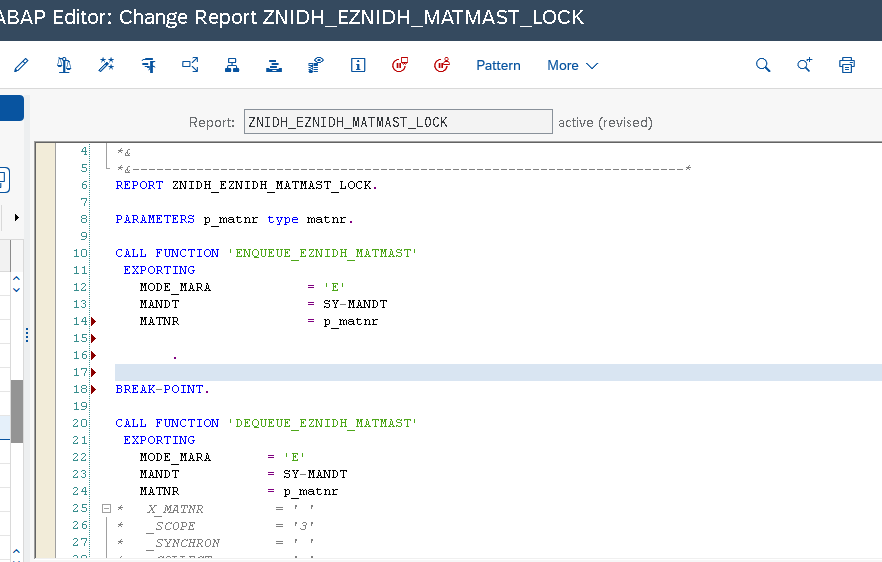
LOCK OBJECTS

\*&---------------------------------------------------------------------\*  
\*& Report ZNIDH\_EZNIDH\_MATMAST\_LOCK  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZNIDH\_EZNIDH\_MATMAST\_LOCK.  
  
PARAMETERS p\_matnr type matnr.  
  
CALL FUNCTION 'ENQUEUE\_EZNIDH\_MATMAST'  
 EXPORTING  
   MODE\_MARA            = 'E'  
   MANDT                = SY-MANDT  
   MATNR                = p\_matnr  
  
       .  
  
BREAK-POINT.  
  
CALL FUNCTION 'DEQUEUE\_EZNIDH\_MATMAST'  
 EXPORTING  
   MODE\_MARA       = 'E'  
   MANDT           = SY-MANDT  
   MATNR           = p\_matnr  
\*   X\_MATNR         = ' '  
\*   \_SCOPE          = '3'  
\*   \_SYNCHRON       = ' '  
\*   \_COLLECT        = ' '

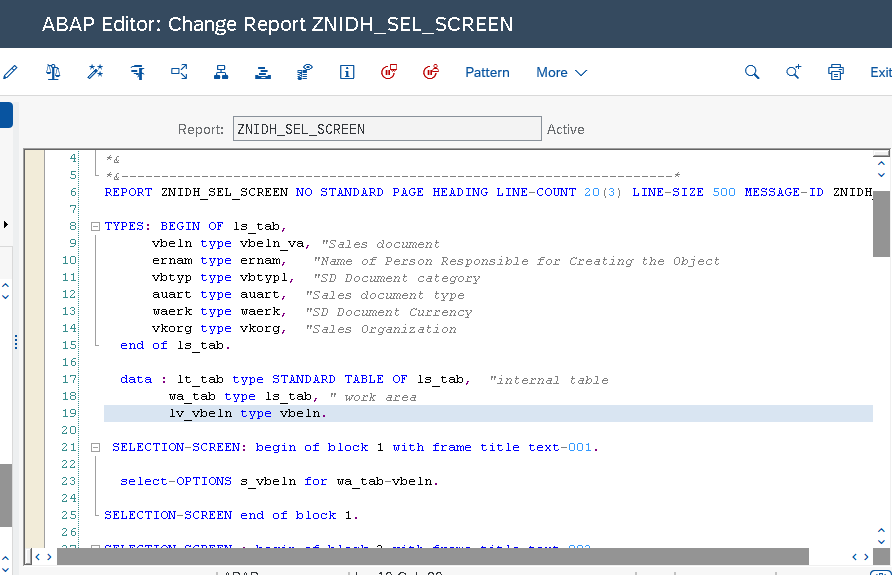
.

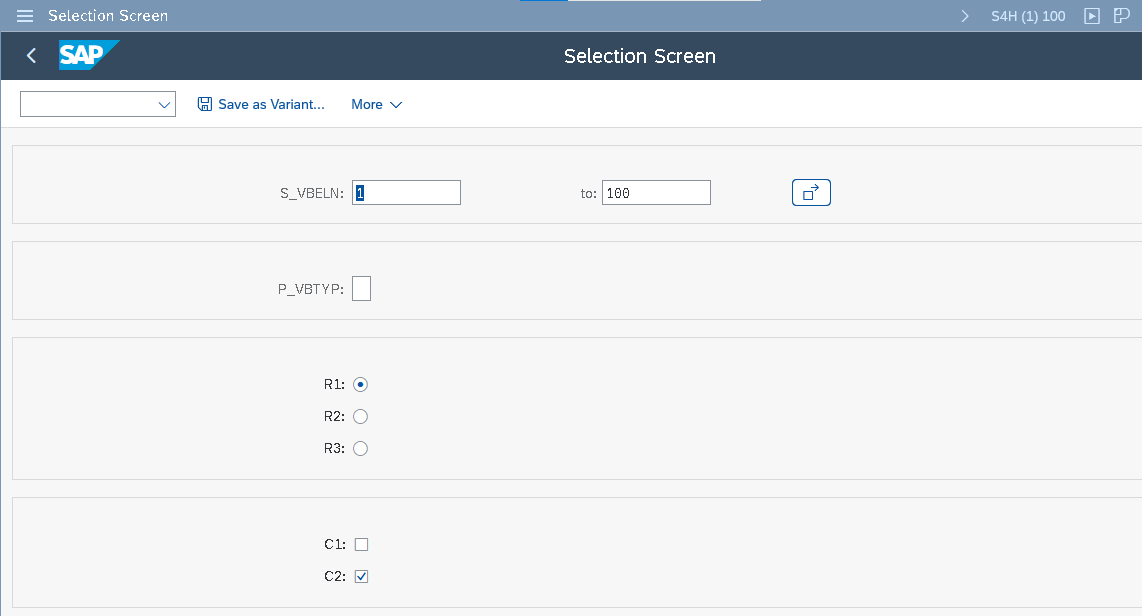


**SELECTION SCREEN**

1. Selection screen ZNIDH\_SEL\_SCREEN

\*&---------------------------------------------------------------------\*  
\*& Report ZNIDH\_SEL\_SCREEN  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZNIDH\_SEL\_SCREEN NO STANDARD PAGE HEADING LINE-COUNT 20(3) LINE-SIZE 500 MESSAGE-ID ZNIDH\_MSG.  
  
TYPES: BEGIN OF ls\_tab,  
      vbeln type vbeln\_va, "Sales document  
      ernam type ernam,   "Name of Person Responsible for Creating the Object  
      vbtyp type vbtypl,  "SD Document category  
      auart type auart,  "Sales document type  
      waerk type waerk,  "SD Document Currency  
      vkorg type vkorg,  "Sales Organization  
  end of ls\_tab.  
  
  data : lt\_tab type STANDARD TABLE OF ls\_tab,  "internal table  
        wa\_tab type ls\_tab, " work area  
        lv\_vbeln type vbeln.  
  
 SELECTION-SCREEN: begin of block 1 with frame title text-001.  
  
  select-OPTIONS s\_vbeln for wa\_tab-vbeln.  
  
SELECTION-SCREEN end of block 1.  
  
SELECTION-SCREEN : begin of block 2 with frame title text-002.  
  PARAMETERS p\_vbtyp type vbtyp.  
 selection-SCREEN end of block 2.  
  
selection-screen : begin of block 3 with frame title text-003.  
  
  PARAMETERS: r1 RADIOBUTTON GROUP rad1,  
              r2 RADIOBUTTON GROUP rad1,  
              r3 RADIOBUTTON GROUP rad1.  
  
selection-screen end of block 3.  
  
selection-screen : begin of block 4 with frame title text-004.  
  
  parameters: c1 as checkbox,  
              c2 as CHECKBOX default 'X'.  
  
selection-screen end of block 4.  
  
INITIALIZATION.  
  
clear: lt\_tab, wa\_tab.  
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.  
   else.  
     MESSAGE e001.  
   endif.  
  
START-OF-SELECTION.  
   perform get\_vbak\_fields.  
end-of-SELECTION.  
   perform disp\_vbak\_fields.  
  
data lv\_count type i.  
   lv\_count = sy-linct - sy-linno.  
skip lv\_count.  
  
top-of-page.  
   uline.  
   write : / 'Sales doc',  
            20 'Name',  
            40 'SD Doc category',  
            55 'SD Doc type',  
            70  'SD Doc currency',  
            85 'Sales Organization'.  
   uline.  
end-of-page.  
write : / 'Page no.-', sy-pagno,  
          'Date-',sy-datum,  
          'Time-',sy-uzeit.  
\*&---------------------------------------------------------------------\*  
\*& Form get\_vbak\_fields  
\*&---------------------------------------------------------------------\*  
FORM get\_vbak\_fields .  
  select vbeln ernam vbtyp auart waerk vkorg  
    into table lt\_tab  
    from vbak  
    where vbeln in s\_vbeln.  
  
    if sy-subrc eq 0.  
      message s000.  
    else.  
      message e001.  
    ENDIF.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form disp\_vbak\_fields  
\*&---------------------------------------------------------------------\*  
FORM disp\_vbak\_fields .  
  loop at lt\_tab into wa\_tab.  
    write : / wa\_tab-vbeln,  
             20 wa\_tab-ernam,  
             40 wa\_tab-vbtyp,  
             55 wa\_tab-auart,  
             70 wa\_tab-waerk,  
             85 wa\_tab-vkorg.  
  ENDLOOP.  
  
ENDFORM.



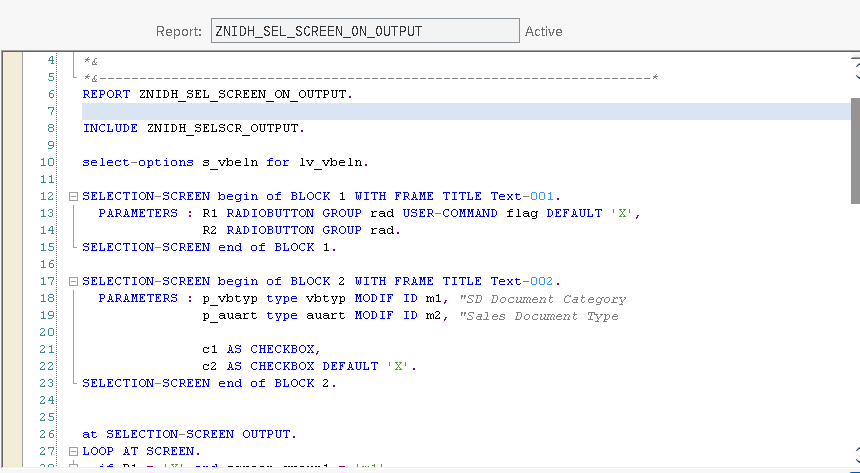


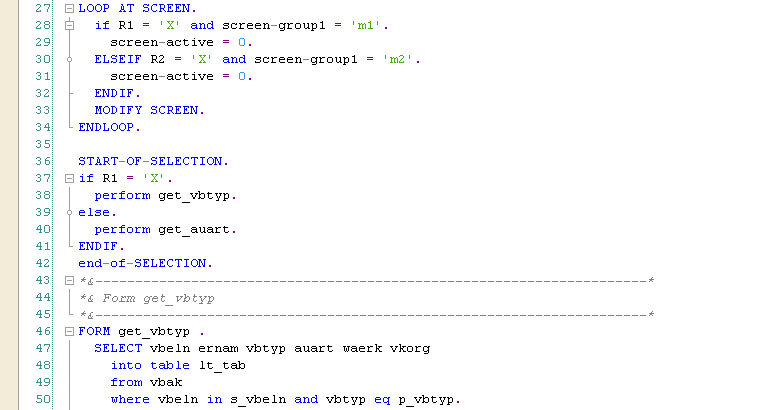
OUTPUT

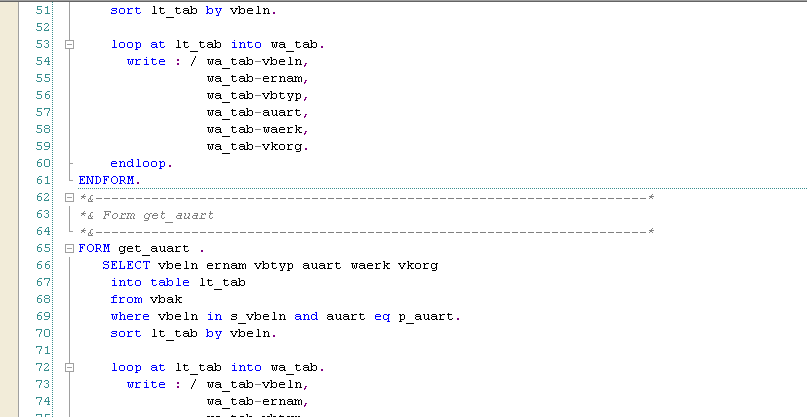


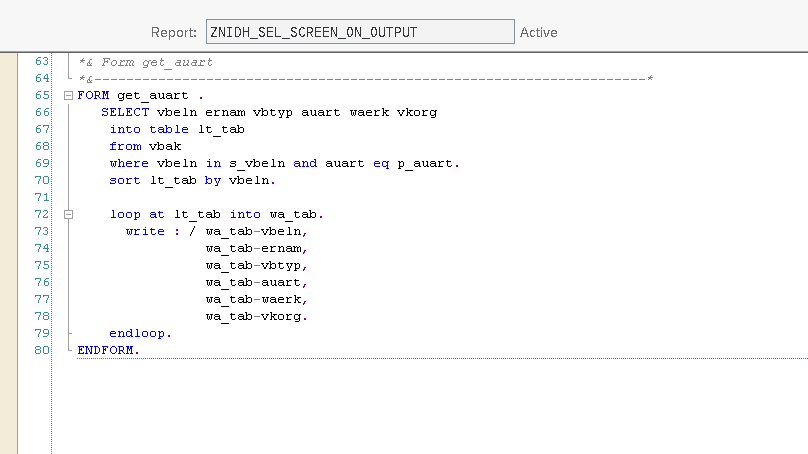
1. SELECTION SCREEN ON OUTPUT

\*&---------------------------------------------------------------------\*  
\*& Report ZNIDH\_SEL\_SCREEN\_ON\_OUTPUT  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZNIDH\_SEL\_SCREEN\_ON\_OUTPUT.  
  
INCLUDE ZNIDH\_SELSCR\_OUTPUT.  
  
select-options s\_vbeln for lv\_vbeln.  
  
SELECTION-SCREEN begin of BLOCK 1 WITH FRAME TITLE Text-001.  
  PARAMETERS : R1 RADIOBUTTON GROUP rad USER-COMMAND flag DEFAULT 'X',  
               R2 RADIOBUTTON GROUP rad.  
SELECTION-SCREEN end of BLOCK 1.  
  
SELECTION-SCREEN begin of BLOCK 2 WITH FRAME TITLE Text-002.  
  PARAMETERS : p\_vbtyp type vbtyp MODIF ID m1, "SD Document Category  
               p\_auart type auart MODIF ID m2, "Sales Document Type  
  
               c1 AS CHECKBOX,  
               c2 AS CHECKBOX DEFAULT 'X'.  
SELECTION-SCREEN end of BLOCK 2.  
  
  
at SELECTION-SCREEN OUTPUT.  
LOOP AT SCREEN.  
  if R1 = 'X' and screen-group1 = 'm1'.  
    screen-active = 0.  
  ELSEIF R2 = 'X' and screen-group1 = 'm2'.  
    screen-active = 0.  
  ENDIF.  
  MODIFY SCREEN.  
ENDLOOP.  
  
START-OF-SELECTION.  
if R1 = 'X'.  
  perform get\_vbtyp.  
else.  
  perform get\_auart.  
ENDIF.  
end-of-SELECTION.  
\*&---------------------------------------------------------------------\*  
\*& Form get\_vbtyp  
\*&---------------------------------------------------------------------\*  
FORM get\_vbtyp .  
  SELECT vbeln ernam vbtyp auart waerk vkorg  
    into table lt\_tab  
    from vbak  
    where vbeln in s\_vbeln and vbtyp eq p\_vbtyp.  
    sort lt\_tab by vbeln.  
  
    loop at lt\_tab into wa\_tab.  
      write : / wa\_tab-vbeln,  
                wa\_tab-ernam,  
                wa\_tab-vbtyp,  
                wa\_tab-auart,  
                wa\_tab-waerk,  
                wa\_tab-vkorg.  
    endloop.  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form get\_auart  
\*&---------------------------------------------------------------------\*  
FORM get\_auart .  
   SELECT vbeln ernam vbtyp auart waerk vkorg  
    into table lt\_tab  
    from vbak  
    where vbeln in s\_vbeln and auart eq p\_auart.  
    sort lt\_tab by vbeln.  
  
    loop at lt\_tab into wa\_tab.  
      write : / wa\_tab-vbeln,  
                wa\_tab-ernam,  
                wa\_tab-vbtyp,  
                wa\_tab-auart,  
                wa\_tab-waerk,  
                wa\_tab-vkorg.  
    endloop.  
ENDFORM.

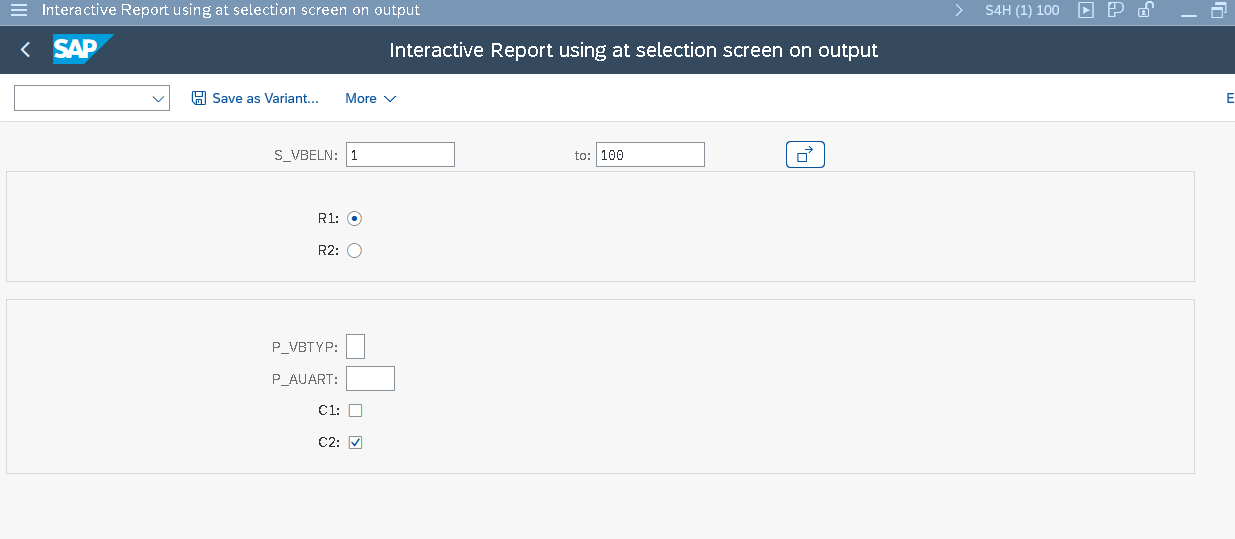








OUTPUT



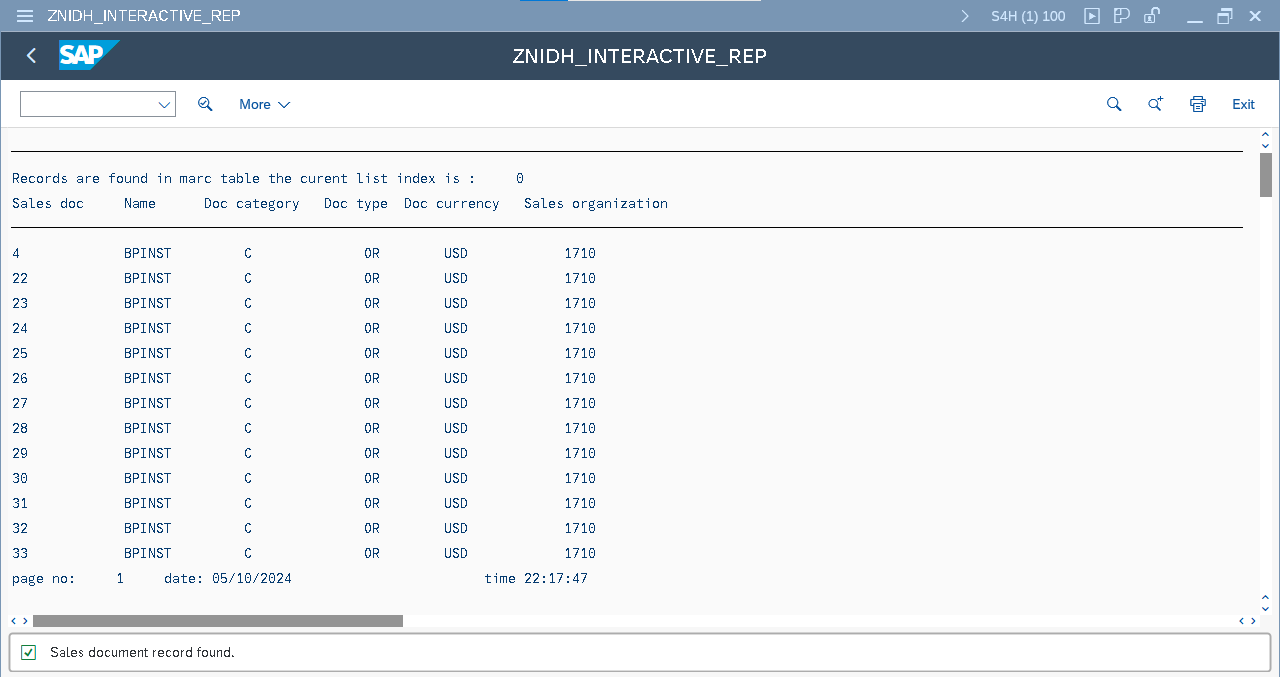
**INTERACTIVE REPORTS**

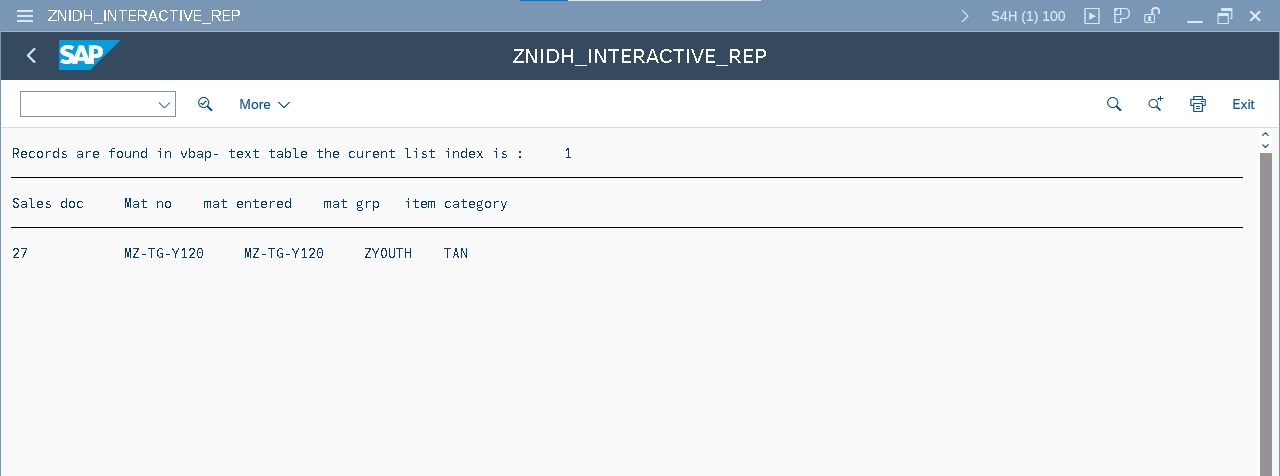
\*&---------------------------------------------------------------------\*  
\*& Report ZNIDH\_INTERACTIVE\_REP  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZNIDH\_INTERACTIVE\_REP NO STANDARD PAGE HEADING LINE-COUNT 20(3) LINE-SIZE 500 MESSAGE-ID ZNIDH\_MSG.  
  
Load-OF-PROGRAM.  
  
INCLUDE ZNIDH\_INCLUDE\_INTERAC\_REP.  
  
SELECT-OPTIONS S\_VBELN FOR WA\_TAB-VBELN OBLIGATORY.  
  
INITIALIZATION.  
clear : lt\_tab , wa\_tab.  
  
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.  
    else.  
       MESSAGE e001.  
    ENDIF .  
  
    START-OF-SELECTION.  
    PERFORM get\_rep\_field.  
    end-of-SELECTION.  
  
    PERFORM disp\_rep\_field.  
  
    data: lv\_count type i.  
    lv\_count = sy-linct - sy-linno.  
    skip lv\_count.  
  
  
    TOP-OF-PAGE.  
    uline.  
    write :/ 'Records are found in marc table', 'the curent list index is :  ', sy-lsind.  
      Write: / 'Sales doc',  
               15 'Name',  
               25 'Doc category',  
               40 'Doc type',  
               50 'Doc currency',  
               65 'Sales organization'.  
    uline.  
    end-of-page.  
  
    write:/ 'page no:', sy-pagno,  
            20 'date:', sy-datum,  
            60 'time', sy-uzeit.  
  
    AT LINE-SELECTION.  
      case sy-lsind.  
        WHEN 1.  
          select vbeln matnr matwa matkl pstyv  
            into table lt\_tab2  
            from vbap  
            where vbeln eq wa\_tab-vbeln.  
  
          loop at lt\_tab2 into wa\_tab2.  
            write : / wa\_tab2-vbeln HOTSPOT,  
                      15 wa\_tab2-matnr,  
                      30 wa\_tab2-matwa,  
                      45 wa\_tab2-matkl,  
                      55 wa\_tab2-pstyv.  
  
            hide wa\_tab2-vbeln.  
  
            ENDLOOP.  
  
  
         WHEN 2.  
          select vbeln abrdt absta  
            into table lt\_tab3  
            from vbag  
            where vbeln eq wa\_tab2-vbeln.  
  
          loop at lt\_tab3 into wa\_tab3.  
            write : / wa\_tab3-vbeln,  
                      15 wa\_tab3-abrdt,  
                      30 wa\_tab3-absta.  
  
  
  
  
            ENDLOOP.  
        endcase.  
  
        TOP-OF-PAGE DURING LINE-SELECTION.  
        case sy-lsind.  
          when 1.  
            write : / 'Records are found in vbap- text table', 'the curent list index is :  ', sy-lsind.  
            uline.  
            write : / 'Sales doc',  
                      15 'Mat no',  
                      25 'mat entered',  
                      40 'mat grp',  
                      50 'item category'.  
            uline.  
  
            when 2.  
            write : / 'Records are found in vbag- text table', 'the curent list index is :  ', sy-lsind.  
            uline.  
            write : / 'Sales doc',  
                      25 'deli sch date',  
                      40 'release status'.  
  
            uline.  
  
         ENDCASE.  
  
\*&---------------------------------------------------------------------\*  
\*& Form get\_rep\_field  
\*&---------------------------------------------------------------------\*  
FORM get\_rep\_field .  
  select vbeln ernam vbtyp auart waerk vkorg  
    into table lt\_tab  
    from vbak  
    where vbeln in s\_vbeln.  
  
    if sy-subrc eq 0.  
      MESSAGE s000.  
    else.  
      MESSAGE e001.  
    ENDIF.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form disp\_rep\_field  
\*&---------------------------------------------------------------------\*  
  
FORM disp\_rep\_field .  
  loop at lt\_tab into wa\_tab.  
    write : / wa\_tab-vbeln,  
              15 wa\_tab-ernam,  
              30 wa\_tab-vbtyp,  
              45 wa\_tab-auart,  
              55 wa\_tab-waerk,  
              70 wa\_tab-vkorg.  
  
    HIDE wa\_tab-vbeln.  
  ENDLOOP.  
  
ENDFORM.

INCLUDE PROGRAM FOR THIS

\*&---------------------------------------------------------------------\*  
\*& Include          ZNIDH\_INCLUDE\_INTERAC\_REP  
\*&---------------------------------------------------------------------\*  
  
types :  BEGIN OF ls\_tab,  
        vbeln type vbeln\_va,  
        ernam type ernam,  
        vbtyp type vbtyp,  
        auart type auart,  
        waerk type waerk,  
        vkorg type vkorg,  
  end of ls\_tab.  
  
  
  data : lt\_tab type STANDARD TABLE OF ls\_tab,  
        wa\_tab type ls\_tab,  
        lv\_vbeln type vbeln.  
  
types: begin of ls\_tab2,  
       vbeln type vbeln\_va,  
       matnr type matnr,  
       matwa type matwa,  
       matkl type matkl,  
       pstyv type pstyv,  
  
  end of ls\_tab2.  
  
  data : lt\_tab2 type STANDARD TABLE OF ls\_tab2,  
        wa\_tab2 type ls\_tab2.  
  
  
types: begin of ls\_tab3,  
      vbeln type vbeln\_va,  
      abrdt type abrdt,  
      absta type absta,  
  
  end of ls\_tab3.  
  
  data : lt\_tab3 type STANDARD TABLE OF ls\_tab3,  
        wa\_tab3 type ls\_tab3.

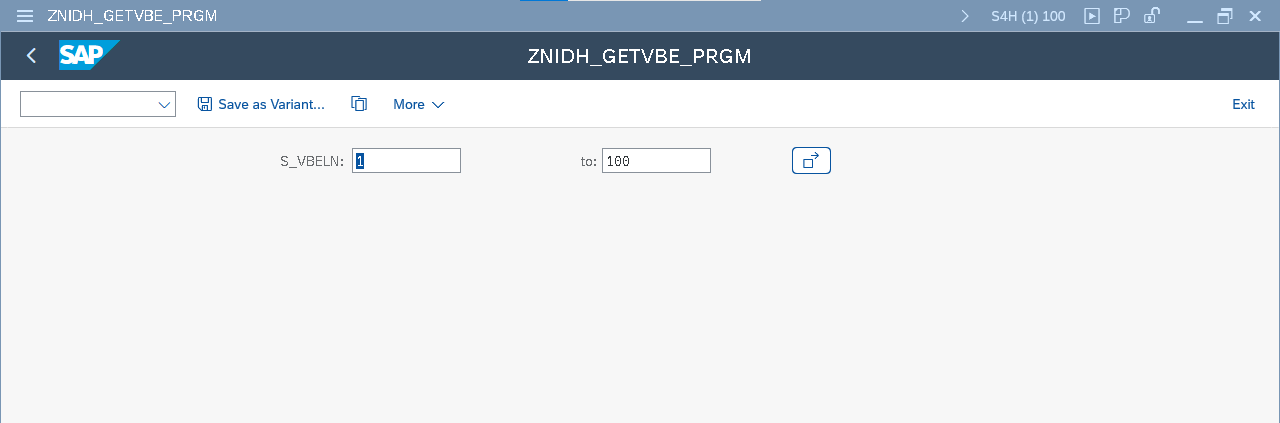






1. GET VBE PROGRAM

\*&---------------------------------------------------------------------\*  
\*& Report ZNIDH\_GETVBE\_PRGM  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZNIDH\_GETVBE\_PRGM.  
  
DATA : lt\_tab type ZNIDH\_TT\_VBE,  
      wa\_tab type ZNIDH\_LT\_VBE.  
  
select-options s\_vbeln for wa\_tab-vbeln.  
  
START-OF-SELECTION.  
  
CALL FUNCTION 'ZNIDH\_GETVBE'  
EXPORTING  
   I\_VBELN1       = s\_vbeln-low"'1'  
   I\_VBELN2       = s\_vbeln-high"'100'  
  TABLES  
    gt\_tab         = lt\_tab  
  
            .  
  
PERFORM disp\_vbe\_data.  
\*&---------------------------------------------------------------------\*  
\*& Form disp\_vbe\_data  
  
FORM disp\_vbe\_data .  
  loop at lt\_tab into wa\_tab.  
    write : / wa\_tab-vbeln,  
              wa\_tab-ernam,  
             wa\_tab-vbtyp,  
              wa\_tab-auart,  
             wa\_tab-waerk,  
              wa\_tab-vkorg.  
  ENDLOOP.  
  
ENDFORM.





ITAB PROGRAMS

USING THE DML OPERATIONS

*\*&---------------------------------------------------------------------\**  
*\*& Report ZNIDH\_ITAB\_PRGMS*  
*\*&---------------------------------------------------------------------\**  
*\*&*  
*\*&---------------------------------------------------------------------\**  
REPORT ZNIDH\_ITAB\_PRGMS.  
  
*\*data: begin of line,*  
*\*  col\_1(20) type c,*  
*\*  col\_2(20) type c,*  
*\*  end of line.*  
*\**  
*\*  data itab like table of  line.*  
*\*  line-col\_1 = 'SAP'.*  
*\*  line-col\_2 = 'ABAP'.*  
*\*  append line to itab.*  
*\*  clear line.*  
*\**  
*\*  refresh itab.*  
*\*  if itab is INITIAL.*  
*\*    write : 'Internal table is empty'.*  
*\*    free itab.*  
*\*  endif.*  
  
*\*\*\*\*\*\*\*\*    INSERT     like standard table  \*\*\*\*\*\*\*\*\**  
  
*\*data : begin of line,*  
*\*  name(20) type c,*  
*\*  age type i,*  
*\*  weight type p DECIMALS 2,*  
*\*  land(20) type c,*  
*\*  end of line.*  
*\**  
*\*types : begin of line,*  
*\*  name(20) type c,*  
*\*  age type i,*  
*\*  weight type p DECIMALS 2,*  
*\*  land(20) type c,*  
*\*  end of line.*  
*\**  
*\*data itab like STANDARD TABLE OF line with NON-UNIQUE key name age weight land.*  
*\**  
*\*line-name = 'Humpty'.*  
*\*line-age = 25.*  
*\*line-weight = '70.00'.*  
*\*line-land = 'UK'.*  
*\*insert line into table itab.*  
*\**  
*\*line-name = 'Dumpty'.*  
*\*line-age = 26.*  
*\*line-weight = '70.00'.*  
*\*line-land = 'UK'.*  
*\*insert line into table itab.*  
*\**  
*\*line-name = 'Jack'.*  
*\*line-age = 23.*  
*\*line-weight = '60.00'.*  
*\*line-land = 'USA'.*  
*\*insert line into table itab.*  
*\**  
*\*\*sort itab by name.*  
*\**  
*\*loop at itab into line.*  
*\*  write : / line-name, line-age, line-weight, line-land.*  
*\*endloop.*  
  
  
  
*\*3  \*\*\*\*\*\*\*    INSERT     like sorted table  \*\*\*\*\*\*\*\*\**  
  
*\*data : begin of line,*  
*\*  name(20) type c,*  
*\*  age type i,*  
*\*  weight type p DECIMALS 2,*  
*\*  land(20) type c,*  
*\*  end of line.*  
*\**  
*\*types : begin of line,*  
*\*  name(20) type c,*  
*\*  age type i,*  
*\*  weight type p DECIMALS 2,*  
*\*  land(20) type c,*  
*\*  end of line.*  
*\**  
*\*data itab like sorted TABLE OF line with NON-UNIQUE key name age weight land.*  
*\**  
*\*line-name = 'Humpty'.*  
*\*line-age = 25.*  
*\*line-weight = '70.00'.*  
*\*line-land = 'UK'.*  
*\*insert line into table itab.*  
*\**  
*\*line-name = 'Dumpty'.*  
*\*line-age = 26.*  
*\*line-weight = '70.00'.*  
*\*line-land = 'UK'.*  
*\*insert line into table itab.*  
*\**  
*\*line-name = 'Jack'.*  
*\*line-age = 23.*  
*\*line-weight = '60.00'.*  
*\*line-land = 'USA'.*  
*\*insert line into table itab.*  
*\**  
*\*loop at itab into line.*  
*\*  write : / line-name, line-age, line-weight, line-land.*  
*\*endloop.*  
  
  
*\*4   \*\*\*\*\*\*\*\* INSERT  \*\*\*\*\*\*\*\**  
  
*\*data: begin of line,*  
*\*  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  \*\*\*\*\*\*\*\*\**  
  
*\*data: begin of line,*  
*\*  col\_1 type i,*  
*\*  col\_2 type i,*  
*\*end of line.*  
*\**  
*\*data itab like table of line.*  
*\**  
*\*DO 3 TIMES.*  
*\*  line-col\_1 = sy-index.*  
*\*  line-col\_2 = sy-index \*\* 2.*  
*\*  append line to itab.*  
*\*ENDDO.*  
*\**  
*\*loop at itab into line.*  
*\*  if sy-tabix = 2.*  
*\*  line-col\_1 = sy-tabix \* 10.*  
*\*  line-col\_2 = ( sy-tabix \* 10 ) \*\* 2.*  
*\*  modify itab from line.*  
*\*  endif.*  
*\**  
*\*endloop.*  
*\**  
*\*loop at itab into line.*  
*\*  write : / sy-tabix,*  
*\*            line-col\_1,*  
*\*            line-col\_2.*  
*\*endloop.*  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*  DELETE  \*\*\*\*\*\*\*\*\*\*\*\*\**  
  
*\*data : BEGIN OF line,*  
*\*  col\_1 type i,*  
*\*  col\_2 type i,*  
*\*end of line.*  
*\**  
*\*data itab like HASHED TABLE OF line with UNIQUE KEY col\_1.*  
*\**  
*\*DO 4 TIMES.*  
*\*  line-col\_1 = sy-index.*  
*\*  line-col\_2 = sy-index \*\* 2.*  
*\*  insert line into table itab.*  
*\*ENDDO.*  
*\**  
*\*line-col\_1 = 1.*  
*\*delete table itab: from line, with table key col\_1 = 3.*  
*\**  
*\*loop at itab into line.*  
*\*  write : / line-col\_1, line-col\_2.*  
*\*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 = 'SAP'.*  
*\*line-col\_2 = '45'.*  
*\*line-col\_3 = 5.*  
*\*collect line into itab.*  
*\*write / sy-tabix.*  
*\**  
*\*line-col\_1 = 'ABAP'.*  
*\*line-col\_2 = '36'.*  
*\*line-col\_3 = 6.*  
*\**  
*\*collect line into itab.*  
*\*write / sy-tabix.*  
*\**  
*\*line-col\_1 = 'UST'.*  
*\*line-col\_2 = '42'.*  
*\*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.*  
  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*       DATA TYPES      \*\*\*\*\*\*\*\*\*\*\*\*\**  
  
*\*types : BEGIN OF ty\_student,*  
*\*  id(5)  type n,*  
*\*  name(10) type c,*  
*\*  end of ty\_student.*  
*\**  
*\**  
*\*  "Data declaration*  
*\**  
*\*data: gwa\_student type ty\_student.*  
*\*data: it type table of ty\_student.*  
*\**  
*\*gwa\_student-ID = 1.*  
*\*gwa\_student-name = 'JOHN'.*  
*\*APPEND gwa\_student to it.*  
*\**  
*\*gwa\_student-ID = 2.*  
*\*gwa\_student-name = 'JIM'.*  
*\*APPEND gwa\_student to it.*  
*\**  
*\*gwa\_student-ID = 1.*  
*\*gwa\_student-name = 'JACK'.*  
*\*APPEND gwa\_student to it.*  
*\**  
*\*sort it by id.*  
*\**  
*\*\*\*\*\*\*#1*  
*\**  
*\*READ TABLE it into gwa\_student index 2.*  
*\**  
*\*if sy-subrc = 0.*  
*\*  WRITE: gwa\_student-ID, gwa\_student-name.*  
*\*else.*  
*\*  write 'No Record Found'.*  
*\*endif.*  
*\**  
*\*\*\*\*\*\* #2*  
*\**  
*\*READ TABLE it into gwa\_student with key name = 'JACK'.*  
*\*if sy-subrc = 0.*  
*\*  write: gwa\_student-ID, gwa\_student-name.*  
*\*else.*  
*\*  write 'No record Found'.*  
*\*endif.*  
*\**  
*\**  
*\*\*\*\*\*\*\*\*  FIELD-SYMBOLS*  
*\**  
*\**  
*\*\*\*\*\*\*\*\*\* 1*  
*\*  data : F1(20) type c.*  
*\**  
*\*  FIELD-SYMBOLS <FS\_A>.*  
*\**  
*\*  F1 = 'SAP ABAP'.*  
*\**  
*\*  ASSIGN F1 to <FS\_A>.*  
*\**  
*\*  write : <FS\_A>.*  
  
  
*\*\*\*\*\*\*\*  2*  
  
data: BEGIN OF line, *"field string*  
      col\_1 type i,  
      col\_2 type i,  
    end of line.  
  
data itab like SORTED TABLE OF line with UNIQUE KEY col\_1.  
  
field-symbols <fs> like line of itab.  
  
do 10 times.  
  line-col\_1 = sy-index.  
  line-col\_2 = 2 \* sy-index.  
  append line to itab.  
enddo.  
  
loop at itab into line.  *"itab to field*  
  write : / line-col\_1, line-col\_2. *"line-weight, line-land.*  
endloop.  
  
  
*\*\*\*\*\*\*\*\*\*\* READ TABLE \*\*\*\*\*\*\*\*\**  
*\* clear <fs>.*  
write : sy-subrc, sy-tabix.  
write: / <fs>-col\_1, <fs>-col\_2.