REPORT ZSAN\_INVOICE\_FORM\_PROG.  
  
INCLUDE ZSAN\_INVOICE\_TOP. "" TOP INCLUDE

data : wa\_outputparams type sfpoutputparams,  
      wa\_outdocparams type sfpdocparams,  
      fnmod type rs38l\_fnam,  
      lv\_vbeln type vbrk-vbeln,  
      lv\_fkdat type vbrk-fkdat,  
      gt\_final type ZSAN\_TAB\_DATA1\_T,  
      lv\_kunnr type kna1-kunnr.  
  
INCLUDE ZSAN\_ATSCREEN.    "" AT SCREEN-SELECTIONS

SELECT-OPTIONS s\_vbeln for lv\_vbeln.  
  
SELECT-OPTIONS s\_fkdat for lv\_fkdat OBLIGATORY.  
  
SELECT-OPTIONS s\_kunnr for lv\_kunnr.  
  
\*\*\*parameters : p\_fkdat type vbrk-fkdat,  
\*\*\*             p\_kunnr type kna1-kunnr.  
  
at SELECTION-SCREEN.  
  select single vbeln from vbrk into lv\_vbeln where vbeln in s\_vbeln and vbrk~fkart = 'F2'.  
    if sy-subrc <> 0.  
      MESSAGE 'Entered Invoice Document invalid , Please Enter valid Invoice Document' type 'E'.  
     endif.  
  
  select single fkdat from vbrk into lv\_fkdat where fkdat in s\_fkdat and vbrk~fkart = 'F2'.  
    if sy-subrc <> 0.  
      MESSAGE 'No Document Found With Given Range' type 'E'.  
     endif.  
  
  select single kunnr from kna1 into lv\_kunnr where kunnr in s\_kunnr.  
    if sy-subrc <> 0.  
      MESSAGE 'Entered Customer Invalid, Please Enter valid Customer' type 'E'.  
     endif.

INCLUDE ZSAN\_MAIN\_CODE.   "" SELECT QUERY

\*&---------------------------------------------------------------------\*  
\*& Include          ZSAN\_MAIN\_CODE  
\*&---------------------------------------------------------------------\*  
  
START-OF-SELECTION.  
select vbrp~aubel , vbrk~vbeln , vbrk~fkdat  
  into table @data(gt\_rkrp)  
  from vbrp inner join vbrk on vbrp~vbeln = vbrk~vbeln  
  inner join vbak on vbrp~aubel = vbak~vbeln  
  where vbrk~vbeln in @s\_vbeln and vbrk~fkart = 'F2' and vbrk~fkdat in @s\_fkdat and vbrk~gbstk = 'C'.  
  
  
  if gt\_rkrp[] is not initial.  
    select vbak~vbeln ,kna1~kunnr, name1, name2, ort01, pstlz, land1 "vbak~vbeln  
      into table @data(gt\_akkn)  
      from kna1 inner join vbak  
      on vbak~kunnr = kna1~kunnr  
      for all ENTRIES IN @gt\_rkrp  
      where vbak~vbeln = @gt\_rkrp-aubel and kna1~kunnr in @s\_kunnr.  
  
      if sy-subrc <> 0.  
        MESSAGE 'No Data Found ' TYPE 'E'.  
      endif.  
  
    select vbeln, posnr, aubel, aupos, vgbel, vgpos, matnr, fkimg, vrkme, netwr, waerk  
      into table @data(gt\_vbrp)  
      from vbrp  
      FOR ALL ENTRIES IN @gt\_rkrp  
      where vbrp~vbeln = @gt\_rkrp-vbeln.  
      if sy-subrc <> 0.  
        MESSAGE 'No Data Found ' TYPE 'E'.  
      endif.  
  
  endif.  
  
  if gt\_rkrp[] is not initial.  
    sort gt\_rkrp by vbeln.  
    delete ADJACENT DUPLICATES FROM gt\_rkrp COMPARING vbeln.  
    sort gt\_akkn by vbeln.  
    delete ADJACENT DUPLICATES FROM gt\_akkn COMPARING vbeln.  
    sort gt\_vbrp by vbeln.  
    delete ADJACENT DUPLICATES FROM gt\_vbrp COMPARING vbeln.  
  
    data(gv\_index) = value syst\_tabix( ).  
  
    loop at gt\_rkrp ASSIGNING FIELD-SYMBOL(<lfs\_rkrp>).  
      append INITIAL LINE TO gt\_final ASSIGNING FIELD-SYMBOL(<gv\_final>).  
     MOVE-CORRESPONDING <lfs\_rkrp> to <gv\_final>.  
  
     read table gt\_akkn ASSIGNING FIELD-SYMBOL(<lfs\_akko>) with key vbeln = <lfs\_rkrp>-aubel.  
     MOVE-CORRESPONDING <lfs\_akko> to <gv\_final>.  
  
     if sy-subrc = 0.  
       gv\_index = sy-tabix.  
       loop at gt\_vbrp ASSIGNING FIELD-SYMBOL(<lfs\_vbrp>) from gv\_index.  
         if <lfs\_rkrp>-vbeln <> <lfs\_vbrp>-vbeln.  
           exit.  
         endif.  
         append INITIAL LINE TO <gv\_final>-items   ASSIGNING FIELD-SYMBOL(<lfs\_item>).  
         MOVE-CORRESPONDING <lfs\_vbrp> to <lfs\_item>.  
       endloop.  
     endif.  
    endloop.  
 endif.  
  
INCLUDE ZSAN\_FM\_CAL.      "" FUNCTION CALLING

\*&---------------------------------------------------------------------\*  
\*& Include          ZSAN\_FM\_CAL  
\*&---------------------------------------------------------------------\*  
  
wa\_outputparams-dest = 'LP01'.  
  wa\_outputparams-device = 'PRINTER'.  
  wa\_outputparams-nodialog = 'X'.  
  wa\_outputparams-preview = 'X'.  
  
  
  
  CALL FUNCTION 'FP\_JOB\_OPEN'  
    CHANGING  
      ie\_outputparams       = WA\_OUTPUTPARAMS  
\*   EXCEPTIONS  
\*     CANCEL                = 1  
\*     USAGE\_ERROR           = 2  
\*     SYSTEM\_ERROR          = 3  
\*     INTERNAL\_ERROR        = 4  
\*     OTHERS                = 5  
            .  
  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
  ENDIF.  
  
  CALL FUNCTION 'FP\_FUNCTION\_MODULE\_NAME'  
    EXPORTING  
      i\_name                     = 'ZSAN\_SALES\_INVOICE\_FORMM'  
    IMPORTING  
     E\_FUNCNAME                 = fnmod  
\*     E\_INTERFACE\_TYPE           =  
\*     EV\_FUNCNAME\_INBOUND        =  
            .  
  
 wa\_outdocparams-langu = 'EN' .  
    wa\_outdocparams-country = 'IN'.  
  
CALL FUNCTION fnmod "'/1BCDWB/SM00001032'  
  EXPORTING  
\*   /1BCDWB/DOCPARAMS        =  
    gt\_final                 = gt\_final  
\* IMPORTING  
\*   /1BCDWB/FORMOUTPUT       =  
\* EXCEPTIONS  
\*   USAGE\_ERROR              = 1  
\*   SYSTEM\_ERROR             = 2  
\*   INTERNAL\_ERROR           = 3  
\*   OTHERS                   = 4  
          .  
IF sy-subrc <> 0.  
\* Implement suitable error handling here  
ENDIF.  
  
  
  
  CALL FUNCTION 'FP\_JOB\_CLOSE'  
\*   IMPORTING  
\*     E\_RESULT             =  
\*   EXCEPTIONS  
\*     USAGE\_ERROR          = 1  
\*     SYSTEM\_ERROR         = 2  
\*     INTERNAL\_ERROR       = 3  
\*     OTHERS               = 4  
            .  
  IF sy-subrc <> 0.  
\* Implement suitable error handling here  
  ENDIF.

INPUT SCREEN AND OUTPUTSCREEN:

