



*Email Attachments:-*

* *

*\*&---------------------------------------------------------------------\**  
*\*& Report  /VWK/78B\_BALANCE\_CONFIRMATION*  
*\*&*  
*\*-----------------------------------------------------------------------\**  
*\* Created by         : Sainath Maadkar(CG)*  
*\* Functional         : Haresh Chache*  
*\* Change Date        : 28/03/2018*  
*\* Change Identifier  : SM 28032018 ++ / SM 28032018 –-*  
*\* Transport Number   : IDTK907212*  
*\* Change Description : New Report For Balance Confirmation*  
*\*----------------------------------------------------------------------\**  
  
REPORT  /vwk/78b\_balance\_confirmation.  
  
TYPE-POOLS : slis, truxs.  
  
TABLES: bsid, kna1, adr6.  
  
TYPES: BEGIN OF ty\_total,  
         kunnr TYPE bsid-kunnr,  
         dmbtr TYPE bsid-dmbtr,  
       END OF ty\_total,  
  
       BEGIN OF ty\_data,  
         kunnr(70),  
         xblnr(35),  
         zuonr(20),  
         belnr(20),  
         blart(20),  
         bldat(20),  
         budat(20),  
         dmbtr(20),  
         waers(20),  
         augbl(20),  
         gsber(15),  
         kkber(20),  
         sgtxt(50),  
         xref1(15),  
       END OF ty\_data,  
  
       BEGIN OF ty\_error,  
*\*         kunnr TYPE kna1-kunnr,*  
*\*         id TYPE char1,*  
         text TYPE char250,  
       END OF ty\_error.  
  
DATA: it\_bsid               TYPE STANDARD TABLE OF bsid,  
      wa\_bsid               TYPE bsid,  
      it\_bsad               TYPE STANDARD TABLE OF bsad,  
      wa\_bsad               TYPE bsad,  
      it\_total              TYPE STANDARD TABLE OF ty\_total,  
      wa\_total              TYPE ty\_total,  
      it\_data               TYPE STANDARD TABLE OF ty\_data,  
      wa\_data               TYPE ty\_data,  
      it\_error              TYPE STANDARD TABLE OF ty\_error,  
      wa\_error              TYPE ty\_error,  
      gt\_mail\_list          TYPE TABLE OF /vwk/78b\_decom,  
      gw\_mail\_list          TYPE /vwk/78b\_decom.  
  
DATA: l\_fname               TYPE rs38l\_fnam,  
      ls\_output\_options     TYPE ssfcompop,  
      ls\_control\_parameters TYPE ssfctrlop,  
      ls\_output\_data        TYPE ssfcrescl,  
      l\_pdf                 TYPE rcl\_bag\_tline,  
      l\_size                TYPE i.  
  
DATA: pdf                   TYPE rcl\_bag\_tline,  
      wx\_document\_bcs       TYPE REF TO cx\_document\_bcs,  
      wx\_send\_req\_bcs       TYPE REF TO cx\_send\_req\_bcs,  
      wx\_address\_bcs        TYPE REF TO cx\_address\_bcs,  
      bcs\_mess              TYPE string.  
  
DATA: attname               TYPE so\_obj\_des, *"Attached File Name*  
      i\_text                TYPE soli\_tab,   *"Body Content*  
      i\_subject             TYPE so\_obj\_des, *"Subject Content*  
      text1(250).                            *"Body Content*  
  
DATA : it\_sort              TYPE slis\_t\_sortinfo\_alv,  
       wa\_sort              TYPE slis\_sortinfo\_alv.  
  
DATA: fname                 TYPE string.  
  
DATA: lines                 TYPE i,  
      gd\_percent            TYPE i,  
      p\_per                 TYPE i,  
      count                 TYPE sy-tabix.  
  
  
SELECTION-SCREEN: BEGIN OF BLOCK b1 WITH FRAME TITLE text-001.  
SELECT-OPTIONS: kunnr FOR bsid-kunnr .  
PARAMETERS: bukrs TYPE bsid-bukrs OBLIGATORY DEFAULT '2680' MODIF ID k1,  
            budat TYPE bsid-budat OBLIGATORY MODIF ID k1.  
SELECTION-SCREEN: END OF BLOCK b1.  
  
SELECTION-SCREEN: BEGIN OF BLOCK b2 WITH FRAME TITLE text-003.  
PARAMETERS: c1 AS CHECKBOX DEFAULT 'X'.  
SELECTION-SCREEN: END OF BLOCK b2.  
  
SELECTION-SCREEN: BEGIN OF BLOCK b3 WITH FRAME TITLE text-002.  
PARAMETERS: o1 RADIOBUTTON GROUP g1 DEFAULT 'X' USER-COMMAND muda\_tela. *"muda\_tela-->Instant Screen Changes*  
PARAMETERS: r1 RADIOBUTTON GROUP g1.  
PARAMETERS: r2 RADIOBUTTON GROUP g1,  
            dwfile LIKE rlgrap-filename MODIF ID d1  
            DEFAULT 'C:\BalanceConfirmation\FolderName1\'.  
SELECTION-SCREEN BEGIN OF LINE.  
SELECTION-SCREEN: COMMENT 1(55) chk1 MODIF ID d1.  
SELECTION-SCREEN END OF LINE.  
SELECTION-SCREEN: END OF BLOCK b3.  
  
  
INITIALIZATION.  
  CONCATENATE sy-datum(4) '03' '31' INTO budat.  
  chk1 = '(Path should be Without File Name & Extension!!)'.  
  
AT SELECTION-SCREEN OUTPUT.  
  PERFORM modify\_screen.  
  
AT SELECTION-SCREEN ON VALUE-REQUEST FOR dwfile.  
  CALL FUNCTION 'KD\_GET\_FILENAME\_ON\_F4'  
    CHANGING  
      file\_name = dwfile.  
  
START-OF-SELECTION.  
  PERFORM get\_data.  
  IF it\_bsid[] IS NOT INITIAL.  
    PERFORM modify\_data.  
    PERFORM local\_download.  
    IF r1 = 'X'.  
      IF it\_error[] IS NOT INITIAL.  
        CALL FUNCTION 'POPUP\_WITH\_TABLE\_DISPLAY'  
          EXPORTING  
            endpos\_col   = 100  
            endpos\_row   = 20  
            startpos\_col = 40  
            startpos\_row = 4  
            titletext    = 'Error/Success Log!!!'  
          TABLES  
            valuetab     = it\_error  
          EXCEPTIONS  
            break\_off    = 1  
            OTHERS       = 2.  
      ELSE.  
        MESSAGE 'Email Successfully Sent to All Customers!' TYPE 'I'.  
      ENDIF.  
    ENDIF.  
    IF c1 = 'X'.  
      PERFORM display.  
    ENDIF.  
    IF o1 = 'X'.  
      IF c1 NE 'X'.  
        MESSAGE 'Please Tick on Checkbox Display ALV Report' TYPE 'I'.  
      ENDIF.  
    ENDIF.  
  ELSE.  
    MESSAGE 'Data Not Found!!' TYPE 'W'.  
  ENDIF.  
  
*\*&---------------------------------------------------------------------\**  
*\*&      Form  modify\_screen*  
*\*&---------------------------------------------------------------------\**  
*\*       text*  
*\*----------------------------------------------------------------------\**  
FORM modify\_screen .  
  
  LOOP AT SCREEN.  
    IF r2 = 'X'.  
      IF screen-group1 = 'D1'.  
        screen-invisible = 0.  
        screen-input     = 1.  
        screen-active    = 1.  
        MODIFY SCREEN.  
        CONTINUE.  
      ENDIF.  
    ELSE.  
      IF screen-group1 = 'D1'.  
        screen-invisible = 1.  
        screen-input     = 0.  
        screen-active    = 0.  
        MODIFY SCREEN.  
        CONTINUE.  
      ENDIF.  
    ENDIF.  
    IF screen-group1 = 'K1'.  
      screen-intensified = 1.  
      MODIFY SCREEN.  
    ENDIF.  
  
  ENDLOOP.  
  
ENDFORM.                    *" MODIFY\_SCREEN*  
  
*\*&---------------------------------------------------------------------\**  
*\*&      Form  GET\_DATA*  
*\*&---------------------------------------------------------------------\**  
FORM get\_data.  
  
  SELECT \* FROM bsid INTO TABLE it\_bsid  
    WHERE bukrs EQ bukrs  
    AND   kunnr IN kunnr  
    AND   budat LE budat.  
*\*  SORT it\_bsid BY kunnr budat.*  
  
  SELECT \* FROM bsad APPENDING TABLE it\_bsid  
    WHERE bukrs EQ bukrs  
    AND   kunnr IN kunnr  
    AND   budat LE budat  
    AND   augdt GE budat.  
  SORT it\_bsid BY kunnr budat.  
  
  IF it\_bsid[] IS NOT INITIAL.  
    SELECT \*  
      FROM /vwk/78b\_decom  
      INTO TABLE gt\_mail\_list  
      FOR ALL ENTRIES IN it\_bsid  
      WHERE dealer EQ it\_bsid-kunnr.  
  
    IF sy-subrc EQ 0.  
      SORT gt\_mail\_list BY dealer.  
    ENDIF.  
  ENDIF.  
  
  
ENDFORM.                    *" GET\_DATA*  
*\*&---------------------------------------------------------------------\**  
*\*&      Form  MODIFY\_DATA*  
*\*&---------------------------------------------------------------------\**  
*\*       text*  
*\*----------------------------------------------------------------------\**  
*\*  -->  p1        text*  
*\*  <--  p2        text*  
*\*----------------------------------------------------------------------\**  
FORM modify\_data.  
  
  LOOP AT it\_bsid INTO wa\_bsid.  
    IF wa\_bsid-shkzg = 'H'.  
      wa\_bsid-dmbtr = wa\_bsid-dmbtr \* -1.  
    ENDIF.  
    wa\_total-kunnr = wa\_bsid-kunnr.  
    wa\_total-dmbtr = wa\_bsid-dmbtr.  
    COLLECT wa\_total INTO it\_total.  
    MODIFY it\_bsid FROM wa\_bsid TRANSPORTING dmbtr.  
    CLEAR: wa\_bsid, wa\_total.  
  ENDLOOP.  
  
ENDFORM.                    *" MODIFY\_DATA*  
  
*\*&---------------------------------------------------------------------\**  
*\*&      Form  display*  
*\*&---------------------------------------------------------------------\**  
*\*       text*  
*\*----------------------------------------------------------------------\**  
FORM display .  
  
  DATA: gt\_fieldcat TYPE slis\_t\_fieldcat\_alv,  
  gs\_layout TYPE slis\_layout\_alv,  
  gt\_sp\_group TYPE slis\_t\_sp\_group\_alv,  
  gt\_events TYPE slis\_t\_event,  
  ls\_layout TYPE slis\_layout\_alv.  
  
  PERFORM create\_fcat USING 'KUNNR' 'Account' 'C500' CHANGING gt\_fieldcat.   *"C500 FOR GREEN COLOR*  
  PERFORM create\_fcat USING 'XBLNR' 'Reference' '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'ZUONR' 'Assignment' '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'BELNR' 'Document No.' 'C410' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'BLART' 'Doc. Type'  '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'BLDAT' 'Doc. Date' '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'BUDAT' 'Posting Date'  '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'DMBTR' 'Amount' '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'WAERS' 'Currency' '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'AUGBL' 'Clrng doc.' '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'GSBER' 'Business place' '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'KKBER' 'Credit Control Area' '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'SGTXT' 'Text' '' CHANGING gt\_fieldcat.  
  PERFORM create\_fcat USING 'XREF1' 'Ref. Key 1' '' CHANGING gt\_fieldcat.  
  
  ls\_layout-colwidth\_optimize = 'X'.  
  ls\_layout-zebra = 'X'.  
  wa\_sort-spos      = 1.  
  wa\_sort-fieldname = 'KUNNR'.  
  wa\_sort-up        = 'X'.  
  wa\_sort-subtot    = 'X'.  
  APPEND wa\_sort TO it\_sort.  
  CALL FUNCTION 'REUSE\_ALV\_GRID\_DISPLAY'  
    EXPORTING  
      i\_callback\_program = sy-repid  
      is\_layout          = ls\_layout  
      it\_fieldcat        = gt\_fieldcat  
      i\_save             = 'X'  
      it\_sort            = it\_sort  
    TABLES  
      t\_outtab           = it\_bsid[].  
  IF sy-subrc <> 0.  
    MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
    WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
  ENDIF.  
  
ENDFORM.                    *" DISPLAY*  
*\**  
*\*\*&---------------------------------------------------------------------\**  
*\*\*&      Form  create\_fcat*  
*\*\*&---------------------------------------------------------------------\**  
*\*\*       text*  
*\*\*----------------------------------------------------------------------\**  
*\*\*      -->P\_FNAME        text*  
*\*\*      -->P\_TITLE        text*  
*\*\*      -->P\_EMPHS        text*  
*\*\*      -->P\_GT\_FIELDCAT  text*  
*\*\*----------------------------------------------------------------------\**  
FORM create\_fcat USING p\_fname  
p\_title  
p\_emphs  
p\_gt\_fieldcat TYPE slis\_t\_fieldcat\_alv.  
  DATA: ls\_fieldcat TYPE slis\_fieldcat\_alv.  
  DATA : lv\_cnt TYPE char2.  
  CLEAR : ls\_fieldcat.  
  ls\_fieldcat-tabname = 'IT\_BSID'.  
  ls\_fieldcat-fieldname = p\_fname.  
  ls\_fieldcat-seltext\_m = p\_title.  
  ls\_fieldcat-col\_pos = lv\_cnt.  
  ls\_fieldcat-emphasize = p\_emphs.  
  IF p\_fname = 'DMBTR'.  
    ls\_fieldcat-do\_sum = 'X'.  
  ENDIF.  
  APPEND ls\_fieldcat TO p\_gt\_fieldcat.  
  lv\_cnt = lv\_cnt + 1.  
ENDFORM. *" CREATE\_FCAT*  
  
  
*\*&---------------------------------------------------------------------\**  
*\*&      Form  LOCAL\_DOWNLOAD*  
*\*&---------------------------------------------------------------------\**  
*\*       text*  
*\*----------------------------------------------------------------------\**  
*\*  -->  p1        text*  
*\*  <--  p2        text*  
*\*----------------------------------------------------------------------\**  
FORM local\_download .  
  
  CLEAR: lines.  
  DESCRIBE TABLE it\_total LINES lines.  
  
  IF it\_total[] IS NOT INITIAL.  
    LOOP AT it\_total INTO wa\_total.  
  
      REFRESH: it\_data.  
      CLEAR: wa\_data.  
      APPEND wa\_data TO it\_data.  
  
      CLEAR: wa\_data.  
      wa\_data-kunnr = 'Customer Code'.  
      wa\_data-xblnr = wa\_total-kunnr.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
  
      DATA: name1 TYPE kna1-name1,  
      ort01 TYPE kna1-ort01.  
      CLEAR: name1, ort01.  
      SELECT SINGLE name1 ort01 FROM kna1 INTO (name1,ort01)  
        WHERE kunnr = wa\_total-kunnr.  
  
      wa\_data-kunnr = 'Name'.  
      wa\_data-xblnr = name1.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
  
      wa\_data-kunnr = 'City'.  
      wa\_data-xblnr = ort01.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
  
      DATA:  cdat TYPE char10,  
             bdat TYPE char10,  
             pdat TYPE char10.  
      CLEAR: cdat.  
      CONCATENATE budat+6(2) '.' budat+4(2) '.' budat(4) INTO cdat.  
      wa\_data-kunnr = 'Outstanding statement as on:'.  
      wa\_data-xblnr = cdat.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
      APPEND wa\_data TO it\_data.  
  
      CLEAR: wa\_data.  
      wa\_data-kunnr = 'Account'.  
      wa\_data-xblnr = 'Reference'.  
      wa\_data-zuonr = 'Assignment'.  
      wa\_data-belnr = 'Document No.'.  
      wa\_data-blart = 'Typ'.  
      wa\_data-bldat = 'Doc.Date'.  
      wa\_data-budat = 'Posting Date'.  
      wa\_data-dmbtr = 'Amtinloc.cur.'.  
      wa\_data-waers = 'LCurr'.  
      wa\_data-augbl = 'Clrng doc'.  
      wa\_data-gsber = 'Business place'.  
      wa\_data-kkber = 'Credit Control Area'.  
      wa\_data-sgtxt = 'Text'.  
      wa\_data-xref1 = 'Ref. Key 1'.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data, wa\_bsid.  
  
      LOOP AT it\_bsid INTO wa\_bsid WHERE kunnr = wa\_total-kunnr.  
        MOVE-CORRESPONDING wa\_bsid TO wa\_data.  
        CLEAR: wa\_data-budat, wa\_data-bldat.  
        CONCATENATE wa\_bsid-budat+6(2) '.' wa\_bsid-budat+4(2) '.' wa\_bsid-budat(4)  
        INTO wa\_data-budat.  
        CONCATENATE wa\_bsid-bldat+6(2) '.' wa\_bsid-bldat+4(2) '.' wa\_bsid-bldat(4)  
        INTO wa\_data-bldat.  
        CALL FUNCTION 'CLOI\_PUT\_SIGN\_IN\_FRONT'  
          CHANGING  
            value = wa\_data-dmbtr.  
        APPEND wa\_data TO it\_data.  
        CLEAR: wa\_data, wa\_bsid.  
      ENDLOOP.  
  
      wa\_data-kunnr = wa\_total-kunnr.  
      wa\_data-sgtxt = 'CLOSING BALANCE'.  
      wa\_data-dmbtr = wa\_total-dmbtr.  
      CALL FUNCTION 'CLOI\_PUT\_SIGN\_IN\_FRONT'  
        CHANGING  
          value = wa\_data-dmbtr.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data, wa\_bsid.  
      APPEND wa\_data TO it\_data.  
  
      wa\_data-kunnr = 'List of document types used:'.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
      wa\_data-kunnr = 'UD - Data takeover AR'.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
      wa\_data-kunnr = 'AB - BALANCE REMAINING AFTER ADJUSTMENT/CLEARING OF DEBITS & CREDITS'.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
      wa\_data-kunnr = 'DR - SALE OF CAR/PARTS INVOICES'.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
      wa\_data-kunnr = 'DP - outgoing inv. Parts'.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
      wa\_data-kunnr = 'DA - DEBIT NOTE ENTRY'.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
      wa\_data-kunnr = 'DG - CREDIT NOTE ENTRY'.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
      wa\_data-kunnr = 'NG - AR Non-GST Posting'.  
      APPEND wa\_data TO it\_data.  
      CLEAR: wa\_data.  
  
      PERFORM dw\_file.  
      PERFORM dw\_letter.  
      REFRESH: it\_data, l\_pdf.  
      CLEAR: wa\_data, wa\_total, l\_size.  
      count = count + 1.  
    ENDLOOP.  
  
    IF r2 = 'X'.  
      MESSAGE 'Files Successfully Downloaded!' TYPE 'I'.  
    ENDIF.  
  ENDIF.  
  
ENDFORM.                    *" LOCAL\_DOWNLOAD*  
  
  
*\*&---------------------------------------------------------------------\**  
*\*&      Form  dw\_file*  
*\*&---------------------------------------------------------------------\**  
*\*       text*  
*\*----------------------------------------------------------------------\**  
FORM dw\_file.  
  
  IF r2 = 'X'.  
    CLEAR: fname.  
    CONCATENATE dwfile wa\_total-kunnr '\_Statement\_' sy-datum '.XLS' INTO fname.  
    CALL FUNCTION 'GUI\_DOWNLOAD'  
      EXPORTING  
        filename                = fname  
        filetype                = 'ASC'  
        append                  = 'X'  
        write\_field\_separator   = 'X'  
      TABLES  
        data\_tab                = it\_data[]  
      EXCEPTIONS  
        file\_write\_error        = 1  
        no\_batch                = 2  
        gui\_refuse\_filetransfer = 3  
        invalid\_type            = 4  
        no\_authority            = 5  
        unknown\_error           = 6  
        header\_not\_allowed      = 7  
        separator\_not\_allowed   = 8  
        filesize\_not\_allowed    = 9  
        header\_too\_long         = 10  
        dp\_error\_create         = 11  
        dp\_error\_send           = 12  
        dp\_error\_write          = 13  
        unknown\_dp\_error        = 14  
        access\_denied           = 15  
        dp\_out\_of\_memory        = 16  
        disk\_full               = 17  
        dp\_timeout              = 18  
        file\_not\_found          = 19  
        dataprovider\_exception  = 20  
        control\_flush\_error     = 21  
        OTHERS                  = 22.  
    IF sy-subrc <> 0.  
    ENDIF.  
  ENDIF.  
  
ENDFORM.                    *" DW\_FILE*  
  
  
*\*&---------------------------------------------------------------------\**  
*\*&      Form  dw\_letter*  
*\*&---------------------------------------------------------------------\**  
*\*       text*  
*\*----------------------------------------------------------------------\**  
FORM dw\_letter .  
  
  CALL FUNCTION 'SSF\_FUNCTION\_MODULE\_NAME'  
    EXPORTING  
      formname           = '/VWK/78B\_BAL\_CONFIRMATION'  
    IMPORTING  
      fm\_name            = l\_fname  
    EXCEPTIONS  
      no\_form            = 1  
      no\_function\_module = 2  
      OTHERS             = 3.  
  IF sy-subrc <> 0.  
    MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
    WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
  ENDIF.  
  
  ls\_control\_parameters-device = 'PRINTER'.  
  ls\_control\_parameters-no\_dialog = 'X'.  
  ls\_control\_parameters-preview = ' '.  
  ls\_control\_parameters-getotf = 'X'.  
  
  ls\_output\_options-tdnoprint  = 'X'.  
  ls\_output\_options-tddest = 'LOCL'.  
  
  CALL FUNCTION l\_fname  
    EXPORTING  
      control\_parameters = ls\_control\_parameters  
      output\_options     = ls\_output\_options  
      user\_settings      = 'X'  
      key\_dt             = budat  
      dmbtr              = wa\_total-dmbtr  
      kunnr              = wa\_total-kunnr  
    IMPORTING  
      job\_output\_info    = ls\_output\_data.  
  IF sy-subrc <> 0.  
  ENDIF.  
  
  CALL FUNCTION 'CONVERT\_OTF'  
    EXPORTING  
      format                = 'PDF'  
      max\_linewidth         = 134  
    IMPORTING  
      bin\_filesize          = l\_size  
    TABLES  
      otf                   = ls\_output\_data-otfdata  
      lines                 = l\_pdf  
    EXCEPTIONS  
      err\_max\_linewidth     = 1  
      err\_format            = 2  
      err\_conv\_not\_possible = 3  
      OTHERS                = 4.  
  
  IF sy-subrc <> 0.  
    MESSAGE i000(/vwk/78b\_znsc) WITH text-041.  
  ENDIF.  
  
  IF r2 = 'X'.  
    CLEAR: fname.  
    CONCATENATE dwfile wa\_total-kunnr '\_Confirmation\_Letter\_' sy-datum '.PDF'  
    INTO fname.  
    cl\_gui\_frontend\_services=>gui\_download(  
      EXPORTING  
        filename = fname  
        filetype = 'BIN'  
      CHANGING  
        data\_tab = l\_pdf[] ).  
  ELSEIF r1 = 'X'.  
    CLEAR: attname.  
    CONCATENATE wa\_total-kunnr '\_' sy-datum INTO attname.  
    PERFORM generate\_mail USING attname.  
  ENDIF.  
  
ENDFORM.                    *" LETTER*  
  
*\*&---------------------------------------------------------------------\**  
*\*&      Form  GENERATE\_MAIL*  
*\*&---------------------------------------------------------------------\**  
*\*       text*  
*\*----------------------------------------------------------------------\**  
*\*      -->P\_ATTNAME  text*  
*\*----------------------------------------------------------------------\**  
FORM generate\_mail     USING    p\_attname.  
  
  DATA: send\_request   TYPE REF TO cl\_bcs.  
  DATA: binary\_content TYPE solix\_tab.  
  DATA: document       TYPE REF TO cl\_document\_bcs.  
  DATA: sender         TYPE REF TO cl\_sapuser\_bcs.  
  DATA: recipient      TYPE REF TO if\_recipient\_bcs.  
  DATA: bcs\_exception  TYPE REF TO cx\_bcs.  
  DATA: sent\_to\_all    TYPE os\_boolean.  
  DATA: xl\_content     TYPE xstring,  
        l\_content      TYPE string.  
  DATA: ip\_subject     TYPE string,  
        s              TYPE string.  
  CLEAR i\_text.  
  CLEAR i\_subject.  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CREATE MAIL SUBJECT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  DATA:  cdate TYPE char10.  
  CLEAR: cdate.  
  CONCATENATE budat+6(2) '.' budat+4(2) '.' budat(4) INTO cdate.  
  IF sy-sysid = 'IDP'.  
    CONCATENATE 'Balance Confirmation as on ' cdate  
    INTO i\_subject RESPECTING BLANKS.  
    CONCATENATE 'Balance Confirmation as on ' cdate  *"Subject Long Text*  
      INTO ip\_subject RESPECTING BLANKS.  
  ELSEIF sy-sysid = 'IDT'.  
    CONCATENATE 'Balance Confirmation as on ' cdate  
    INTO i\_subject RESPECTING BLANKS.  
    CONCATENATE 'Balance Confirmation as on ' cdate  *"Subject Long Text*  
    INTO ip\_subject RESPECTING BLANKS.  
  ELSEIF sy-sysid = 'IDQ'.  
    CONCATENATE 'TEST-IDQ-Balance Confirmation as on ' cdate  
  INTO i\_subject RESPECTING BLANKS.  
    CONCATENATE 'TEST-IDQ-Balance Confirmation as on ' cdate  
  INTO ip\_subject RESPECTING BLANKS.                                                    *"Subject Long Text*  
  ENDIF.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CREATE MAIL SUBJECT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CREATE MAIL BODY\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  text1 = 'Dear Sir,'. APPEND text1 TO i\_text.  
  text1 = ''. APPEND text1 TO i\_text.  
  text1 = 'Please find attached the balance confirmation letter attached for your dealership customer code.'. APPEND text1 TO i\_text.  
  text1 = ''. APPEND text1 TO i\_text.  
  text1 = 'You are requested to sign the same digitally and send it back to us on below email id'. APPEND text1 TO i\_text.  
  text1 = 'For VW & Lambo Brand – Ganesh.sapar@volkswagen.co.in'. APPEND text1 TO i\_text.  
  text1 = 'For Audi & Porsche Brand – Narendra.majgaonkar@volkswagen.co.in'. APPEND text1 TO i\_text.  
  text1 = ''. APPEND text1 TO i\_text.  
  text1 = 'If you want to highlight any differences,'. APPEND text1 TO i\_text.  
  text1 = 'please send a separate statement duly reconciled with our balances.'. APPEND text1 TO i\_text.  
  text1 = ''. APPEND text1 TO i\_text.  
  text1 = ''. APPEND text1 TO i\_text.  
  text1 = 'Thanks'. APPEND text1 TO i\_text.  
  text1 = 'Volkswagen Group Sales India Pvt. Ltd.'. APPEND text1 TO i\_text.  
  
  LOOP AT it\_data INTO wa\_data.  
    CALL FUNCTION 'CLOI\_PUT\_SIGN\_IN\_FRONT'  
      CHANGING  
        value = wa\_data-dmbtr.  
    CONCATENATE wa\_data-kunnr wa\_data-xblnr wa\_data-zuonr wa\_data-belnr wa\_data-blart wa\_data-bldat  
                wa\_data-budat wa\_data-dmbtr wa\_data-waers wa\_data-augbl wa\_data-gsber  
                wa\_data-kkber wa\_data-sgtxt wa\_data-xref1  
                cl\_abap\_char\_utilities=>cr\_lf  
      INTO s SEPARATED BY cl\_abap\_char\_utilities=>horizontal\_tab.  
    CONCATENATE l\_content s INTO l\_content.  
  ENDLOOP.  
  REPLACE ALL OCCURRENCES OF '–' IN l\_content WITH ''.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CREATE MAIL BODY\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CREATE MAIL DOCUMENT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  TRY.  
      send\_request = cl\_bcs=>create\_persistent( ).  
  
      document = cl\_document\_bcs=>create\_document(  
                      i\_type    = 'RAW'  
                      i\_text    = i\_text  
                      i\_subject = i\_subject ).  
    CATCH cx\_send\_req\_bcs INTO wx\_send\_req\_bcs.  
      bcs\_mess = wx\_send\_req\_bcs->get\_text( ).  
  
*\*\*\*Set Subject Long Text.*  
      DATA: ob\_bcs TYPE REF TO cl\_bcs.  
      CALL METHOD ob\_bcs->set\_message\_subject  
        EXPORTING  
          ip\_subject = ip\_subject.  
  ENDTRY.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*CREATE MAIL DOCUMENT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*BEGIN OF -- GENERATE EXCEL FILE ATTACHMENT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* XLS*  
  CALL FUNCTION 'HR\_KR\_STRING\_TO\_XSTRING'  
    EXPORTING  
      codepage\_to      = '1100'  
      unicode\_string   = l\_content  
      out\_len          = 0  
    IMPORTING  
      xstring\_stream   = xl\_content  
    EXCEPTIONS  
      invalid\_codepage = 1  
      invalid\_string   = 2  
      OTHERS           = 3.  
  
  IF sy-subrc NE 0.  
    CALL FUNCTION 'SCMS\_STRING\_TO\_XSTRING'  
      EXPORTING  
        text   = l\_content  
      IMPORTING  
        buffer = xl\_content  
      EXCEPTIONS  
        failed = 1  
        OTHERS = 2.  
  ENDIF.  
  CALL FUNCTION 'SCMS\_XSTRING\_TO\_BINARY'  
    EXPORTING  
      buffer     = xl\_content  
    TABLES  
      binary\_tab = binary\_content.  
  
  CLEAR: p\_attname.  
  CONCATENATE wa\_total-kunnr 'Statement' sy-datum INTO p\_attname  
  SEPARATED BY '\_'.  
  TRY.  
      CALL METHOD document->add\_attachment  
        EXPORTING  
          i\_attachment\_type    = 'XLS'  
          i\_attachment\_subject = p\_attname  
          i\_att\_content\_hex    = binary\_content.  
    CATCH cx\_document\_bcs INTO wx\_document\_bcs.  
      bcs\_mess = wx\_document\_bcs->get\_text( ).  
  ENDTRY.  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*END OF -- GENERATE EXCEL FILE ATTACHMENT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*BEGIN OF -- GENERATE PDF FILE ATTACHMENT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* PDF*  
  DATA:   l\_soli\_tab  TYPE soli\_tab.  
  REFRESH: l\_soli\_tab.  
  CALL FUNCTION 'SX\_TABLE\_LINE\_WIDTH\_CHANGE'  
    TABLES  
      content\_in  = l\_pdf  
      content\_out = l\_soli\_tab.  
  
  CLEAR: xl\_content.  
  CALL FUNCTION 'SCMS\_BINARY\_TO\_XSTRING'  
    EXPORTING  
      input\_length = l\_size  
    IMPORTING  
      buffer       = xl\_content  
    TABLES  
      binary\_tab   = l\_soli\_tab.  
  
  REFRESH: binary\_content.  
  CALL FUNCTION 'SCMS\_XSTRING\_TO\_BINARY'  
    EXPORTING  
      buffer     = xl\_content  
    TABLES  
      binary\_tab = binary\_content.  
  
  CLEAR: p\_attname.  
  CONCATENATE wa\_total-kunnr 'Confirmation\_Letter' sy-datum INTO p\_attname  
  SEPARATED BY '\_'.  
  TRY.  
      CALL METHOD document->add\_attachment  
        EXPORTING  
          i\_attachment\_type    = 'PDF'  
          i\_attachment\_subject = p\_attname  
          i\_att\_content\_hex    = binary\_content.  
      CALL METHOD send\_request->set\_document( document ).  
    CATCH cx\_document\_bcs INTO wx\_document\_bcs.  
      bcs\_mess = wx\_document\_bcs->get\_text( ).  
  ENDTRY.  
*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*END OF -- GENERATE PDF FILE ATTACHMENT\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
  
*\*\*\*\*\*\*\*\*\*\*ASSIGN SENDER MAIL ADDRESS\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  DATA: lo\_sender TYPE REF TO if\_sender\_bcs VALUE IS INITIAL.  
  DATA: temp\_syb\_dest TYPE adr6-smtp\_addr. *"VALUE 'dealers.payment.details@volkswagen.co.in'.*  
  
  DATA: addsmtp TYPE bapiadsmtp OCCURS 0 WITH HEADER LINE,  
        return TYPE bapiret2 OCCURS 0,  
        wa\_addsmtp LIKE LINE OF addsmtp.  
  
  CLEAR: wa\_addsmtp.  
  REFRESH: return, addsmtp.  
  CALL FUNCTION 'BAPI\_USER\_GET\_DETAIL'  
    EXPORTING  
      username = sy-uname  
    TABLES  
      return   = return  
      addsmtp  = addsmtp.  
  IF sy-subrc = 0 AND addsmtp[] IS NOT INITIAL.  
    READ TABLE addsmtp INTO wa\_addsmtp INDEX 1.  
    temp\_syb\_dest = wa\_addsmtp-e\_mail.  
  ENDIF.  
  IF temp\_syb\_dest IS NOT INITIAL.  
    TRY.  
        lo\_sender = cl\_cam\_address\_bcs=>create\_internet\_address( i\_address\_string = temp\_syb\_dest  
         i\_address\_name = temp\_syb\_dest ).  
  
        send\_request->set\_sender(  
        EXPORTING  
        i\_sender = lo\_sender ).  
      CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
        bcs\_mess = wx\_address\_bcs->get\_text( ).  
    ENDTRY.  
  ENDIF.  
*\*\*\*\*\*\*\*\*\*\*ASSIGN SENDER MAIL ADDRESS\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
  
*\*\*\*\*\*\*\*\*\*\*ASSIGN MULTIPLE RECIPENT MAIL ADDRESS\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
*\*  SELECT SINGLE \* FROM kna1 WHERE kunnr = wa\_total-kunnr.*  
*\*  SELECT SINGLE \* FROM adr6 WHERE addrnumber = kna1-adrnr.*  
*\*  IF adr6-smtp\_addr IS NOT INITIAL.*  
*\*    TRY.*  
*\*        recipient = cl\_cam\_address\_bcs=>create\_internet\_address( adr6-smtp\_addr ).*  
*\*        CALL METHOD send\_request->add\_recipient*  
*\*          EXPORTING*  
*\*            i\_recipient = recipient*  
*\*            i\_express   = 'X'.*  
*\*      CATCH cx\_address\_bcs INTO wx\_address\_bcs.*  
*\*        bcs\_mess = wx\_address\_bcs->get\_text( ).*  
*\*    ENDTRY.*  
*\*  ENDIF.*  
  
  CLEAR: gw\_mail\_list.  
  READ TABLE gt\_mail\_list INTO gw\_mail\_list WITH KEY dealer = wa\_total-kunnr BINARY SEARCH.  
  IF sy-subrc = 0.  
    IF gw\_mail\_list-email1 IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email1 ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email2 IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email2 ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email3 IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email3 ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email4 IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email4 ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email5 IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email5 ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email\_am IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email\_am ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email\_am\_parts IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email\_am\_parts ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email\_rm IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email\_rm ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email\_rm\_parts IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email\_rm\_parts ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email1\_int IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email1\_int ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email2\_int IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email2\_int ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
    IF gw\_mail\_list-email3\_int IS NOT INITIAL.  
      TRY.  
          recipient = cl\_cam\_address\_bcs=>create\_internet\_address( gw\_mail\_list-email3\_int ).  
          CALL METHOD send\_request->add\_recipient  
            EXPORTING  
              i\_recipient = recipient  
              i\_express   = 'X'.  
        CATCH cx\_address\_bcs INTO wx\_address\_bcs.  
          bcs\_mess = wx\_address\_bcs->get\_text( ).  
      ENDTRY.  
    ENDIF.  
  
  ENDIF.  
*\*\*\*\*\*\*\*\*\*\*ASSIGN MULTIPLE RECIPENT MAIL ADDRESS\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
  p\_per = 100.  
  DATA: pv TYPE char20.  
  CONCATENATE 'Sending.' wa\_total-kunnr INTO pv.  
  PERFORM progress\_bar USING pv  
  count  
  lines  
  p\_per.  
  
*\*\*\*\*\*\*\*\*\*\*SEND EMAIL\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  CALL METHOD send\_request->set\_status\_attributes( i\_requested\_status = 'N' ).  
  
  IF temp\_syb\_dest IS NOT INITIAL AND (  
     gw\_mail\_list-email1 IS NOT INITIAL OR gw\_mail\_list-email2 IS NOT INITIAL  
    OR gw\_mail\_list-email3 IS NOT INITIAL OR gw\_mail\_list-email4 IS NOT INITIAL  
    OR gw\_mail\_list-email5 IS NOT INITIAL OR gw\_mail\_list-email\_am IS NOT INITIAL  
    OR gw\_mail\_list-email\_am\_parts IS NOT INITIAL OR gw\_mail\_list-email\_rm IS NOT INITIAL  
    OR gw\_mail\_list-email\_rm\_parts IS NOT INITIAL OR gw\_mail\_list-email1\_int IS NOT INITIAL  
    OR gw\_mail\_list-email2\_int IS NOT INITIAL OR gw\_mail\_list-email3\_int IS NOT INITIAL ).  
    TRY.  
        CALL METHOD send\_request->send(  
          EXPORTING  
            i\_with\_error\_screen = 'X'  
          RECEIVING  
            result              = sent\_to\_all ).  
        COMMIT WORK.  
        IF sy-subrc = 0.  
*\*          CLEAR: wa\_error.*  
*\*          CONCATENATE 'Email Successfully Send for customer 'wa\_total-kunnr*  
*\*                       INTO wa\_error-text SEPARATED BY space.*  
*\*          APPEND wa\_error TO it\_error.*  
        ELSE.  
          CLEAR: wa\_error.  
          CONCATENATE 'Error While sending Email for Custmer-' wa\_total-kunnr  
                       INTO wa\_error-text SEPARATED BY space.  
          APPEND wa\_error TO it\_error.  
        ENDIF.  
      CATCH cx\_send\_req\_bcs INTO wx\_send\_req\_bcs.  
        bcs\_mess = wx\_send\_req\_bcs->get\_text( ).  
    ENDTRY.  
  ELSE.  
    CLEAR: wa\_error.  
    CONCATENATE 'Sender/Recipient Email Id not found for' wa\_total-kunnr  
                 INTO wa\_error-text SEPARATED BY space.  
    APPEND wa\_error TO it\_error.  
  ENDIF.  
*\*\*\*\*\*\*\*\*\*\*SEND EMAIL\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\**  
  
ENDFORM.                    *" GENERATE\_MAIL*  
  
*\*&---------------------------------------------------------------------\**  
*\*&      Form  progress\_bar*  
*\*&---------------------------------------------------------------------\**  
*\*       text*  
*\*----------------------------------------------------------------------\**  
*\*      -->P\_VALUE    text*  
*\*      -->P\_TABIX    text*  
*\*      -->P\_NLINES   text*  
*\*      -->P\_PER      text*  
*\*----------------------------------------------------------------------\**  
FORM progress\_bar USING p\_value  
p\_tabix  
p\_nlines  
p\_per.  
  DATA: w\_text(40),  
  w\_percentage TYPE p,  
  w\_percent\_char(3).  
  IF p\_nlines = 0.  
    w\_percentage = p\_tabix \* p\_per.  
  ELSE.  
    w\_percentage = ( p\_tabix / p\_nlines ) \* p\_per.  
  ENDIF.  
  w\_percent\_char = w\_percentage.  
  SHIFT w\_percent\_char LEFT DELETING LEADING ' '.  
*\*  CONCATENATE p\_value w\_percent\_char '% Complete' INTO w\_text.*  
  w\_text = p\_value.  
*\* This check needs to be in otherwise when looping around big tables*  
*\* SAP will re-display indicator too many times causing report to run*  
*\* very slow. (No need to re-display same percentage anyway)*  
  IF w\_percentage GT gd\_percent OR p\_tabix EQ 1.  
    CALL FUNCTION 'SAPGUI\_PROGRESS\_INDICATOR'  
      EXPORTING  
        percentage = w\_percentage  
        text       = w\_text.  
    gd\_percent = w\_percentage.  
  ENDIF.  
ENDFORM. *" PROGRESS\_BAR* *ATE\_MAIL*