\*&---------------------------------------------------------------------\*  
\*& Report ZTA003\_EMAIL\_WITH\_LINK  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT zta003\_email\_with\_link.  
  
DATA : gt\_text TYPE bcsy\_text. "Table for body  
DATA : gs\_text LIKE LINE OF gt\_text. "work area for message body  
  
DATA binary\_content TYPE solix\_tab.  
DATA size           TYPE so\_obj\_len.  
  
PARAMETERS : p\_ebeln TYPE ebeln.  
PARAMETERS : p\_email TYPE char80.  
  
  
START-OF-SELECTION.  
  PERFORM send\_email.  
\*&---------------------------------------------------------------------\*  
\*& Form send\_email  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM send\_email .  
\*For Email.  
  DATA lo\_send\_request   TYPE REF TO cl\_bcs.  
  DATA lo\_document       TYPE REF TO cl\_document\_bcs.  
  DATA lo\_recipient      TYPE REF TO if\_recipient\_bcs.  
  DATA lo\_bcs\_exception  TYPE REF TO cx\_bcs.  
  DATA lv\_sent\_to\_all    TYPE os\_boolean.  
  DATA lv\_mailto         TYPE ad\_smtpadr.  
  DATA lv\_sub            TYPE so\_obj\_des.  
  
  DATA: lo\_ex\_ref   TYPE REF TO cx\_root,  
        lv\_msg\_text TYPE string.  
  
  CONSTANTS: lc\_htm  TYPE so\_obj\_tp VALUE 'HTM',  
             lc\_tab  TYPE c VALUE cl\_bcs\_convert=>gc\_tab,  
             lc\_crlf TYPE c VALUE cl\_bcs\_convert=>gc\_crlf.  
  
  
  PERFORM create\_pdf\_file.  
  
  TRY.  
\*     Create persistent send request ------------------------  
      lo\_send\_request = cl\_bcs=>create\_persistent( ).  
\*     Create Email body using HTML.  
      PERFORM create\_email\_body.  
  
\*     Create document object from internal table with text  
      lv\_sub = 'PO Details'.  
      lo\_document = cl\_document\_bcs=>create\_document(  
        i\_type    = lc\_htm  
        i\_text    = gt\_text  
        i\_subject = lv\_sub ).                               "#EC NOTEXT  
  
  
\*    add the PDF as attachment to document object  
      lo\_document->add\_attachment(  
        i\_attachment\_type    = 'PDF'                        "#EC NOTEXT  
        i\_attachment\_subject = 'PO'                 "#EC NOTEXT  
        i\_attachment\_size    = size  
        i\_att\_content\_hex    = binary\_content ).  
  
  
  
\*     Add document object to send request  
      lo\_send\_request->set\_document( lo\_document ).  
  
\*--------- add recipient (e-mail address) -----------------------  
\*      LOOP AT s\_email.  
\*     Create recipient object  
      lv\_mailto = p\_email.  
      TRY."Here Recipient is email input p\_email  
          lo\_recipient = cl\_cam\_address\_bcs=>create\_internet\_address( lv\_mailto ). ""#EC CI\_SEL\_NESTED  
        CATCH  cx\_address\_bcs INTO lo\_ex\_ref.  
          lv\_msg\_text = lo\_ex\_ref->get\_text( ).  
\*      Catch exception here  
      ENDTRY.  
      TRY.  
\*      Add recipient object to send request  
          lo\_send\_request->add\_recipient( lo\_recipient ).  
        CATCH  cx\_send\_req\_bcs INTO lo\_ex\_ref.  
          lv\_msg\_text = lo\_ex\_ref->get\_text( ).  
\*      Catch exception here  
      ENDTRY.  
\*      ENDLOOP.  
\*     ---------- send document ---------------------------------------  
      lv\_sent\_to\_all = lo\_send\_request->send( i\_with\_error\_screen = 'X' ).  
      COMMIT WORK.  
      IF lv\_sent\_to\_all IS INITIAL.  
        MESSAGE i500(sbcoms) WITH lv\_mailto.  
      ELSE.  
        MESSAGE s022(so).  
      ENDIF.  
  
\*   ------------ exception handling ----------------------------------  
    CATCH cx\_bcs INTO lo\_bcs\_exception.  
      MESSAGE i865(so) WITH lo\_bcs\_exception->error\_type.  
  ENDTRY.  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form create\_email\_body  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM create\_email\_body .  
  DATA: lv\_link TYPE string,  
        lv\_po   TYPE string.  
  
  
  lv\_link = 'https://fiorih4s.dot.hawaii.gov/sap/bc/ui2/flp?sap-client=420&sap-language=EN#ZPO-display?PoNr='.  
  lv\_link = lv\_link && p\_ebeln.  
  
  lv\_po =  '<a href="' && lv\_link &&'">' && p\_ebeln && '</a>'.  
  
  
  gs\_text-line = '<html><body>'.  
  APPEND gs\_text TO gt\_text.  
  
  gs\_text-line = '<p>Dear Sir/ Madam,</p>'.  
  APPEND gs\_text TO gt\_text.  
  
  
  gs\_text-line = '<p>'.  
  APPEND gs\_text TO gt\_text.  
  gs\_text-line = 'The Purchase Order - ' && lv\_po && ', has been created'.  
  APPEND gs\_text TO gt\_text.  
  gs\_text-line = '</p>'.  
  APPEND gs\_text TO gt\_text.  
  
  gs\_text-line = '<p></p>'.  
  APPEND gs\_text TO gt\_text.  
  
\*  gs\_text-line = '<p><u>SUMMARY:</u></p>'.  
\*  APPEND gs\_text TO gt\_text.  
\*  
\*  gs\_text-line = '<p></p>'.  
\*  APPEND gs\_text TO gt\_text.  
\*  
\*  
\*  gs\_text-line = '<table border = "1" cellpadding = "2" style="color:#000">'.  
\*  APPEND gs\_text TO gt\_text.  
\*  
\*  gs\_text-line = '<tr bgcolor="#f3f9ff" >'  
\*    && '<th>' && 'Total Records'    && '</th>'  
\*    && '<th>' && 'Successfully updated' && '</th>'  
\*    && '<th>' && 'Error'  && '</th></tr>'.  
\*  APPEND gs\_text TO gt\_text.  
\*  
\*  gs\_text-line = '<tr bgcolor="#f3f9ff" >'  
\*    && '<th>' && |{ gv\_total   ALPHA = OUT }|  && '</th>'  
\*    && '<th>' && |{ gv\_success ALPHA = OUT }| && '</th>'  
\*    && '<th>' && |{ gv\_error   ALPHA = OUT }| && '</th></tr>'.  
\*  APPEND gs\_text TO gt\_text.  
\*  
\*  gs\_text-line = '</table>'.  
\*  APPEND gs\_text TO gt\_text.  
  
  
  gs\_text-line = '<p><br></br></p>'.  
  APPEND gs\_text TO gt\_text.  
  
\* Thank you!  
  gs\_text-line = '<p>Thank You!</p>'.  
  APPEND gs\_text TO gt\_text.  
  
  gs\_text-line = '</body></html>'.  
  APPEND gs\_text TO gt\_text.  
  
  
  
ENDFORM.  
\*&---------------------------------------------------------------------\*  
\*& Form create\_pdf\_file  
\*&---------------------------------------------------------------------\*  
\*& text  
\*&---------------------------------------------------------------------\*  
\*& -->  p1        text  
\*& <--  p2        text  
\*&---------------------------------------------------------------------\*  
FORM create\_pdf\_file .  
  DATA: lv\_fname TYPE rs38l\_fnam.  
  
  "Internal Tables.  
  DATA: lt\_otfdata        TYPE ssfcrescl,  
        lt\_binary\_content TYPE solix\_tab,  
        lt\_text           TYPE bcsy\_text,  
        lt\_pdf\_tab        TYPE STANDARD TABLE OF tline,  
        lt\_otf            TYPE STANDARD TABLE OF itcoo.  
  
  "Work Areas  
  DATA: ls\_ctrlop TYPE ssfctrlop,  
        ls\_outopt TYPE ssfcompop.  
  
  "Variables  
  DATA: lv\_bin\_filesize TYPE so\_obj\_len,  
        lv\_sent\_to\_all  TYPE os\_boolean,  
        lv\_bin\_xstr     TYPE xstring,  
        lv\_string\_text  TYPE string.  
  
  
  
  "Control Parameters  
  ls\_ctrlop-getotf    = 'X'.  
  ls\_ctrlop-no\_dialog = 'X'.  
  ls\_ctrlop-preview   = space.  
  
  "Output Options  
  ls\_outopt-tdnoprev  = 'X'.  
  ls\_outopt-tddest    = 'LP01'.  
  ls\_outopt-tdnoprint = 'X'.  
  
  CALL FUNCTION 'SSF\_FUNCTION\_MODULE\_NAME'  
    EXPORTING  
      formname           = 'ZMMSF\_PO\_PROCUREMENT'  
    IMPORTING  
      fm\_name            = lv\_fname  
    EXCEPTIONS  
      no\_form            = 1  
      no\_function\_module = 2  
      OTHERS             = 3.  
  IF sy-subrc = 0.  
    CALL FUNCTION lv\_fname  
      EXPORTING  
        control\_parameters = ls\_ctrlop  
        output\_options     = ls\_outopt  
        iv\_pcard           = p\_ebeln  
        USER\_SETTINGS      = space  
      IMPORTING  
        job\_output\_info    = lt\_otfdata  
      EXCEPTIONS  
        formatting\_error   = 1  
        internal\_error     = 2  
        send\_error         = 3  
        user\_canceled      = 4  
        OTHERS             = 5.  
    IF sy-subrc = 0.  
      lt\_otf[] = lt\_otfdata-otfdata[].  
  
      CALL FUNCTION 'CONVERT\_OTF'  
        EXPORTING  
          format                = 'PDF'  
        IMPORTING  
          bin\_filesize          = lv\_bin\_filesize  
          bin\_file              = lv\_bin\_xstr  
        TABLES  
          otf                   = lt\_otf[]  
          lines                 = lt\_pdf\_tab[]  
        EXCEPTIONS  
          err\_max\_linewidth     = 1  
          err\_format            = 2  
          err\_conv\_not\_possible = 3  
          OTHERS                = 4.  
      IF sy-subrc = 0.  
        CALL FUNCTION 'SCMS\_XSTRING\_TO\_BINARY'  
          EXPORTING  
            buffer     = lv\_bin\_xstr  
          TABLES  
            binary\_tab = lt\_binary\_content.  
  
        binary\_content[] = lt\_binary\_content[].  
        size = lv\_bin\_filesize.  
  
      ENDIF.  
    ENDIF.  
  
  ELSE.  
    MESSAGE ID sy-msgid TYPE sy-msgty NUMBER sy-msgno  
            WITH sy-msgv1 sy-msgv2 sy-msgv3 sy-msgv4.  
  ENDIF.  
  
ENDFORM.