\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
REPORT zms\_license NO STANDARD PAGE HEADING MESSAGE-ID zms.  
  
TABLES: usr02, usr06, usr21, adrp, adr6, tutyp, somlreci1, sscrfields.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\* type definition:  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
TYPE-POOLS: slis.  
  
TYPES: BEGIN OF ty\_usr02,  
         mandt TYPE mandt,  
         bname TYPE xubname,  
         ustyp TYPE xuustyp,  
         gltgv TYPE xugltgv, " user valid from  
         gltgb TYPE xugltgb, " user valid to  
       END OF ty\_usr02.  
  
TYPES: BEGIN OF ty\_usr06,  
         mandt    TYPE mandt,  
         bname    TYPE xubname,  
         lic\_type TYPE usertypen,  
       END OF ty\_usr06.  
  
TYPES: BEGIN OF ty\_usr21,  
         mandt      TYPE mandt,  
         bname      TYPE xubname,  
         persnumber TYPE ad\_persnum,  
         addrnumber TYPE ad\_addrnum,  
       END OF ty\_usr21.  
  
TYPES: BEGIN OF ty\_adrp,  
         client     TYPE mandt,  
         persnumber TYPE ad\_persnum,  
         date\_from  TYPE ad\_date\_fr,  
         nation     TYPE ad\_nation,  
         name\_first TYPE ad\_namefir,  
         name\_last  TYPE ad\_namelas,  
       END OF ty\_adrp.  
  
TYPES: BEGIN OF ty\_final,  
         mandt      TYPE mandt,  
         bname      TYPE xubname,  
         lic\_type   TYPE usertypen,  
         utyptext   TYPE utyptext,  
         name\_first TYPE ad\_namefir,  
         name\_last  TYPE ad\_namelas,  
         smtp\_addr  TYPE ad\_smtpadr,  
         gltgv      TYPE xugltgv, " user valid from  
         gltgb      TYPE xugltgb, " user valid to  
       END OF ty\_final.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\* data definition:  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
  
DATA gv\_repid TYPE sy-repid.  
DATA gv\_layout TYPE slis\_layout\_alv.  
DATA gt\_fcat TYPE slis\_t\_fieldcat\_alv.  
DATA gs\_fcat LIKE LINE OF gt\_fcat.  
DATA gv\_smtp\_addr TYPE ad\_smtpadr.  
DATA gv\_utyptext TYPE utyptext.  
DATA: gt\_usr02 TYPE STANDARD TABLE OF ty\_usr02,  
      gs\_usr02 TYPE ty\_usr02,  
      gt\_usr06 TYPE STANDARD TABLE OF ty\_usr06,  
      gs\_usr06 TYPE ty\_usr06,  
      gt\_usr21 TYPE STANDARD TABLE OF ty\_usr21,  
      gs\_usr21 TYPE ty\_usr21,  
      gt\_adrp  TYPE STANDARD TABLE OF ty\_adrp,  
      gs\_adrp  TYPE ty\_adrp,  
      gt\_final TYPE STANDARD TABLE OF ty\_final,  
      gs\_final TYPE ty\_final.  
DATA vcomm TYPE sscrfields-ucomm.  
DATA gt\_email TYPE TABLE OF string WITH HEADER LINE.  
DATA gr\_table TYPE REF TO cl\_salv\_table.  
  
CONSTANTS:  
  gc\_tab  TYPE c VALUE cl\_bcs\_convert=>gc\_tab,  
  gc\_crlf TYPE c VALUE cl\_bcs\_convert=>gc\_crlf.  
  
DATA send\_request   TYPE REF TO cl\_bcs.  
DATA document       TYPE REF TO cl\_document\_bcs.  
DATA recipient      TYPE REF TO if\_recipient\_bcs.  
DATA bcs\_exception  TYPE REF TO cx\_bcs.  
  
DATA main\_text      TYPE bcsy\_text.  
DATA binary\_content TYPE solix\_tab.  
DATA size           TYPE so\_obj\_len.  
DATA sent\_to\_all    TYPE os\_boolean.  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\* Initialization:  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
INITIALIZATION.  
  INCLUDE z\_check\_program.  "standard check of reports  
  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
\* Selection Screen:  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  SELECTION-SCREEN BEGIN OF BLOCK 001 WITH FRAME TITLE TEXT-001.  
  SELECTION-SCREEN SKIP.  
  SELECT-OPTIONS s\_mandt FOR usr06-mandt.  
  SELECT-OPTIONS s\_lic FOR usr06-lic\_type.  
  SELECT-OPTIONS s\_ustyp FOR usr02-ustyp.  
  SELECTION-SCREEN END OF BLOCK 001.  
  
  SELECTION-SCREEN BEGIN OF BLOCK 002 WITH FRAME TITLE TEXT-002.  
  
  SELECTION-SCREEN BEGIN OF LINE.  
  SELECTION-SCREEN: COMMENT 2(25) TEXT-004.  
  PARAMETERS p\_email AS CHECKBOX USER-COMMAND chk. " DEFAULT 'X'.  
  SELECTION-SCREEN END OF LINE.  
  
  SELECTION-SCREEN BEGIN OF LINE.  
  SELECTION-SCREEN: COMMENT 2(25) TEXT-005 MODIF ID 001.  
  PARAMETERS: p\_recept TYPE ad\_smtpadr MODIF ID 001.  
  SELECTION-SCREEN END OF LINE.  
  
  SELECTION-SCREEN BEGIN OF LINE.  
  SELECTION-SCREEN: COMMENT 2(75) TEXT-006 MODIF ID 001.  
  SELECTION-SCREEN END OF LINE.  
  SELECTION-SCREEN END OF BLOCK 002.  
  
  
AT SELECTION-SCREEN OUTPUT.  
  
\*------- send per mail or not ? ----------------------  
  IF p\_email = 'X'.  
    PERFORM enable\_fields USING '001'.  
  ELSE.  
    PERFORM disable\_fields USING '001'.  
  ENDIF.  
  
\*------- start of selection -> get data ----------------------  
START-OF-SELECTION.  
  
  PERFORM get\_data.  
  
\*------- end of selection -> display data ----------------------  
END-OF-SELECTION.  
  
  PERFORM show\_data USING gt\_final.  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  enable\_fields  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->UV\_GROUP   text  
\*----------------------------------------------------------------------\*  
FORM enable\_fields USING pv\_group.  
  DATA: BEGIN OF ls\_screen.  
          INCLUDE STRUCTURE screen.  
        DATA: END OF ls\_screen.  
  
  CLEAR ls\_screen.  
  
  LOOP AT SCREEN INTO ls\_screen.  
    IF ls\_screen-group1   = pv\_group.  
      ls\_screen-active    = '1'.  
      ls\_screen-invisible = '0'.  
      ls\_screen-input     = '1'.  
      ls\_screen-output    = '1'.  
      MODIFY SCREEN FROM ls\_screen.  
    ENDIF.  
  ENDLOOP.  
  
ENDFORM.                    "enable\_fields  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  disable\_fields  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*      -->UV\_GROUP   text  
\*----------------------------------------------------------------------\*  
FORM disable\_fields USING pv\_group.  
  DATA: BEGIN OF ls\_screen.  
          INCLUDE STRUCTURE screen.  
        DATA: END OF ls\_screen.  
  
  CLEAR ls\_screen.  
  
  LOOP AT SCREEN INTO ls\_screen.  
    IF ls\_screen-group1   = pv\_group.  
      ls\_screen-active    = '0'.  
      ls\_screen-invisible = '1'.  
      ls\_screen-input     = '0'.  
      ls\_screen-output    = '0'.  
      MODIFY SCREEN FROM ls\_screen.  
    ENDIF.  
  ENDLOOP.  
ENDFORM.                    "disable\_fields  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  get\_data  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM get\_data.  
  
\*-------- Selecting data from usr02 table ------------------------  
  SELECT mandt bname ustyp gltgv gltgb  
    FROM usr02 CLIENT SPECIFIED INTO TABLE gt\_usr02  
    WHERE mandt IN s\_mandt AND ustyp IN s\_ustyp.  
  IF sy-subrc EQ 0.  
    SORT gt\_usr02 BY mandt.  
  ENDIF.  
  
  
  LOOP AT gt\_usr02 INTO gs\_usr02.  
    CLEAR: gs\_usr06, gs\_usr21, gs\_adrp, gs\_final.  
\*-------- getting data from table usr06 ------------------------  
    SELECT SINGLE mandt bname lic\_type  
      FROM usr06 CLIENT SPECIFIED INTO gs\_usr06  
      WHERE mandt = gs\_usr02-mandt  
        AND bname = gs\_usr02-bname.  
  
      CHECK gs\_usr06-lic\_type IN s\_lic. " check license type in selection screen  
  
      CLEAR gv\_utyptext.  
      SELECT SINGLE utyptext FROM tutyp INTO gv\_utyptext  
      WHERE usertyp = gs\_usr06-lic\_type AND langu EQ sy-langu.  
      IF sy-subrc EQ 0.  
        gs\_final-utyptext = gv\_utyptext.  
      ENDIF.  
      MOVE-CORRESPONDING gs\_usr06 TO gs\_final.  
  
  
    SELECT SINGLE mandt bname persnumber addrnumber  
      FROM usr21 INTO gs\_usr21  
      WHERE bname = gs\_usr02-bname.  
  
    IF sy-subrc = 0. " if there is data in usr21 table  
      CLEAR gv\_smtp\_addr.  
      SELECT SINGLE smtp\_addr FROM adr6 INTO gv\_smtp\_addr  
      WHERE addrnumber = gs\_usr21-addrnumber AND persnumber = gs\_usr21-persnumber.  
      IF sy-subrc EQ 0. " if there is data in adr6 table  
        gs\_final-smtp\_addr = gv\_smtp\_addr.  
      ENDIF.  
  
      SELECT SINGLE client persnumber date\_from nation name\_first name\_last  
      FROM adrp INTO gs\_adrp  
      WHERE persnumber = gs\_usr21-persnumber.  
      IF sy-subrc = 0. " if there is data in adrp table  
        gs\_final-name\_first = gs\_adrp-name\_first.  
        gs\_final-name\_last = gs\_adrp-name\_last.  
      ENDIF.  
    ENDIF.  
  
    MOVE-CORRESPONDING gs\_usr02 TO gs\_final.  
    APPEND gs\_final TO gt\_final.  
  
  ENDLOOP.  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  
  
ENDFORM.  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  sent\_email  
\*&---------------------------------------------------------------------\*  
\*       text  
\*----------------------------------------------------------------------\*  
\*  -->  p1        text  
\*  <--  p2        text  
\*----------------------------------------------------------------------\*  
FORM sent\_email.  
  
  DATA: lv\_xml      TYPE xstring,  
        lv\_xml\_type TYPE salv\_bs\_constant.  
  
  DATA lp\_xls\_size TYPE so\_obj\_len.  
  DATA xls\_content TYPE solix\_tab.  
  DATA lv\_text TYPE c LENGTH 100.  
  DATA lv\_address  TYPE  ad\_smtpadr.  
  
\*-------- split the email address by comma and put it in table ------------------------  
  SPLIT p\_recept AT ',' INTO TABLE gt\_email.  
  
\*-------- convert the alv data into xml form (xstring) ------------------------  
  lv\_xml\_type = if\_salv\_bs\_xml=>c\_type\_xlsx.  
  lv\_xml = gr\_table->to\_xml( xml\_type = lv\_xml\_type ).  
  lp\_xls\_size = xstrlen( lv\_xml ).  
  xls\_content = cl\_document\_bcs=>xstring\_to\_solix( ip\_xstring = lv\_xml ).  
  
  
  TRY.  
  
\*-------- create persistent send request ------------------------  
      send\_request = cl\_bcs=>create\_persistent( ).  
  
\*-------- create Mail Body Text ------------------------  
      APPEND 'Dear User,' TO main\_text.  
      APPEND ' ' TO main\_text.  
      CONCATENATE 'Please find attached the SAP User License which are not allowed to be in system ' sy-sysid INTO lv\_text.  
      APPEND lv\_text TO main\_text.  
      APPEND ' ' TO main\_text.  
      APPEND 'Best Regards' TO main\_text.  
      APPEND ' ' TO main\_text.  
      APPEND 'Your System administrator' TO main\_text.  
  
\*     ---------- create document ---------------------------------------  
      document = cl\_document\_bcs=>create\_document(  
            i\_type = 'RAW'  
            i\_text = main\_text  
            i\_subject = 'SAP License Monitoring' ).  
  
\*     ---------- add attachment ---------------------------------------  
      document->add\_attachment(  
            i\_attachment\_type = 'EXT'   " EXT stands for Extension for All PC application  
            i\_attachment\_subject = 'USER\_LICENSE.xlsx'  
            i\_attachment\_size = lp\_xls\_size  
            i\_att\_content\_hex = xls\_content ).  
  
      send\_request->set\_document( document ).  
  
\*     ---------- email recipients ---------------------------------------  
      LOOP AT gt\_email.  
        lv\_address = gt\_email.  
  
        recipient = cl\_cam\_address\_bcs=>create\_internet\_address( i\_address\_string = lv\_address ).  
        send\_request->add\_recipient( i\_recipient = recipient ).  
      ENDLOOP.  
  
\*     ---------- block email delivery notification ---------------------------------------  
      CALL METHOD send\_request->set\_status\_attributes  
        EXPORTING  
          i\_requested\_status = 'N'.  " N stand for turnoff  
  
\*     ---------- send document ---------------------------------------  
      sent\_to\_all = send\_request->send( i\_with\_error\_screen = 'X' ).  
  
      IF sent\_to\_all = 'X'.  
        MESSAGE s022(so).  
      ENDIF.  
      COMMIT WORK.  
  
\*   ------------ exception handling ----------------------------------  
  
    CATCH cx\_bcs INTO bcs\_exception.  
      MESSAGE i865(so) WITH bcs\_exception->error\_type.  
  ENDTRY.  
  
ENDFORM.  
  
  
\*&---------------------------------------------------------------------\*  
\*&      Form  SHOW\_DATA  
\*&---------------------------------------------------------------------\*  
FORM show\_data USING ut\_data TYPE STANDARD TABLE.  
  
  DATA lr\_functions TYPE REF TO cl\_salv\_functions.  
  DATA lr\_columns TYPE REF TO cl\_salv\_columns\_table.  
  DATA lr\_column TYPE REF TO cl\_salv\_column\_table.  
  TRY.  
      cl\_salv\_table=>factory(  
        IMPORTING  
          r\_salv\_table = gr\_table  
        CHANGING  
          t\_table = ut\_data ).  
  
    CATCH cx\_salv\_msg.  
  
  ENDTRY.  
  
\* --------------Set functions ---------------------------  
  lr\_functions = gr\_table->get\_functions( ).  
  lr\_functions->set\_all( 'X' ).  
  
  
\* -------------Get columns object ---------------------------  
  lr\_columns = gr\_table->get\_columns( ).  
  
\* ------------Optimize column width ---------------------------  
  lr\_columns->set\_optimize( ).  
  
\* ----------- Set column attributs ---------------------------  
  lr\_column ?= lr\_columns->get\_column( 'SMTP\_ADDR' ).  
  lr\_column->set\_output\_length( '50' ).  
  
\* ------------if alv as well as email address is not empty -> send email --------------  
  IF NOT gt\_final IS INITIAL.  
    IF p\_email = 'X'.  
      PERFORM sent\_email.  
    ENDIF.  
  ENDIF.  
  
\* -----------Show ALV grid -----------------------------------------  
  gr\_table->display( ).  
  
ENDFORM.                    " SHOW\_DATA