\*&---------------------------------------------------------------------\*  
\*& Report ZSWAP\_81  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSWAP\_81.  
  
TABLES lfa1.  
  
DATA : lv\_client TYPE MANDT ,  
      lv\_city TYPE ORT01\_GP,  
      lv\_district TYPE ORT02\_GP.  
  
  
SELECT MANDT , ORT01,ORT02  
  INTO (@lv\_client , @lv\_city , @lv\_district)  
  FROM lfa1.  
  
  WRITE :/ lv\_client,lv\_city,lv\_district.  
  
ENDSELECT .  
  
---------------------------------------------------------------------------------------------------  
  
  
\*&---------------------------------------------------------------------\*  
\*& Report ZSWAP\_81\_02  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSWAP\_81.  
  
TABLES lfa1.  
  
DATA : lv\_client TYPE LIFNR ,  
      lv\_city TYPE ORT01\_GP,  
      lv\_district TYPE ORT02\_GP.  
  
SELECT-OPTIONS S\_client FOR lv\_client.  
PARAMETERS p\_city TYPE ORT01\_GP.  
  
SELECT LIFNR ORT01 ORT02  
  INTO (lv\_client , lv\_city , lv\_district)  
  FROM lfa1  
  WHERE LIFNR in s\_client and ORT01 eq p\_city.  
  
  WRITE :/ lv\_client,lv\_city,lv\_district.  
  
ENDSELECT .  
  
  
---------------------------------------------------------------------------------------------------  
  
  
  
\*&---------------------------------------------------------------------\*  
\*& Report ZSWAP\_81\_CNTRLS  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT zswap\_81\_cntrls.  
  
\*DO 5 TIMES .  
\*  WRITE /'do statement'.  
\*ENDDO.  
  
DATA s TYPE i VALUE 1 .  
  
  
DO .  
  IF sy-index EQ 2.  
    WRITE 'it is two'.  
  ENDIF.  
  WRITE sy-index.  
  CHECK sy-index EQ 2.  
  WRITE sy-index.  
ENDDO.  
  
  
---------------------------------------------------------------------------------------------------  
  
  
\*&---------------------------------------------------------------------\*  
\*& Report ZSWAP\_81\_JOINS  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSWAP\_81\_JOINS.  
  
TYPES : BEGIN OF ls\_tab,  
\*  DATA FROM EKPO TABLE  
        EBELN TYPE EBELN,  
        EBELP TYPE EBELP,  
        LOEKZ TYPE LOEKZ,  
\* DATA FROM EKKO TABLE  
        BUKRS TYPE BUKRS,  
        BSTYP TYPE BSTYP,  
        MATNR TYPE MATNR,  
        END OF ls\_tab.  
  
DATA : lt\_tab TYPE STANDARD TABLE OF ls\_tab,  
       wa\_tab TYPE ls\_tab.  
  
START-OF-SELECTION.  
SELECT EKPO~EBELN  
       EKPO~EBELP  
       EKPO~LOEKZ  
  
\*       DATA FROM EKKO  
      EKKO~BUKRS  
      EKKO~BSTYP  
INTO TABLE lt\_tab  
    FROM EKPO  
  
    INNER JOIN EKKO  
    ON EKPO~EBELN eq EKKO~EBELN.  
\*    WHERE vbeln IN s\_vbeln.  
  
  IF sy-subrc EQ 0.  
    WRITE : 'retreival succes ', sy-dbcnt.  
  ELSE.  
    WRITE : 'failure retrieval', sy-dbcnt.  
  ENDIF.  
  
END-OF-SELECTION.  
  LOOP AT lt\_tab INTO wa\_tab.  
    WRITE : / wa\_tab-EBELN,  
    wa\_tab-EBELP,  
    wa\_tab-LOEKZ,  
    wa\_tab-BUKRS,  
    wa\_tab-BSTYP.  
  ENDLOOP.  
  
  
  
---------------------------------------------------------------------------------------------------  
  
  
\*&---------------------------------------------------------------------\*  
\*& Report ZSWAP\_81\_02  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSWAP\_81.  
  
TABLES ZSWAP\_UST\_DB.  
  
DATA : lv\_Name TYPE ZSWAP\_UST\_DB-NAME ,  
      lv\_city TYPE ZSWAP\_UST\_DB-PLACE,  
      lv\_animal TYPE ZSWAP\_UST\_DB-ANIMAL.  
  
SELECT-OPTIONS S\_Name FOR lv\_Name.  
PARAMETERS p\_city TYPE ZSWAP\_UST\_DB-PLACE.  
  
SELECT NAME PLACE ANIMAL  
  INTO (lv\_Name , lv\_city , lv\_animal )  
  FROM ZSWAP\_UST\_DB  
  WHERE NAME in s\_Name and PLACE eq p\_city.  
  
  WRITE :/ lv\_Name,lv\_city,lv\_animal.  
  
ENDSELECT .  
  
  
---------------------------------------------------------------------------------------------------  
  
  
\*&---------------------------------------------------------------------\*  
\*& Report ZSWAP\_81\_MYTABLE\_2  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT zswap\_81\_mytable\_2.  
\*TABLES kna1.  
\*  
\*DATA: lt\_tab type kna1 occurs 0 WITH HEADER LINE.  
\*DATA lv\_data type kunnr.  
\*  
\*SELECT-OPTIONS s\_kunnr for lv\_data.  
\*PARAMETERS p\_para TYPE kna1-land1.  
\*  
\*SELECT  mandt kunnr land1 name1  
\*  INTO lt\_tab  
\*  FROM kna1  
\*  WHERE kunnr in s\_kunnr and land1 eq p\_para.  
\*  
\*APPEND lt\_tab TO lt\_tab."h to b"  
\*  
\*ENDSELECT.  
\*  
\*LOOP at lt\_tab into lt\_tab. "b to h"  
\*  WRITE: / lt\_tab-kunnr,   " h-f  
\*          lt\_tab-name1,  
\*          lt\_tab-land1.  
\*ENDLOOP.  
  
  
\* Without header  
  
TABLES kna1.  
  
\*TYPES : BEGIN OF ls\_tab,  
\*        kunnr TYPE kunnr,  
\*        name1 TYPE name1\_gp,  
\*        land1 TYPE land1\_gp,  
\*        ort01 TYPE ort01\_gp,  
\*       END OF ls\_tab.  
DATA : ls\_tab TYPE kna1 OCCURS 0 WITH HEADER LINE.  
  
DATA lv\_data TYPE kunnr.  
SELECT-OPTIONS s\_kunnr FOR lv\_data.  
PARAMETERS p\_land1 TYPE kna1-land1.  
  
SELECT mandt kunnr land1 name1  INTO TABLE ls\_tab  
      FROM kna1  
      WHERE kunnr IN s\_kunnr AND land1 EQ p\_land1.  
\*  APPEND ls\_tab TO ls\_tab.  
  
LOOP AT ls\_tab INTO ls\_tab.  
  WRITE : / ls\_tab-kunnr ,  
            ls\_tab-land1,  
            ls\_tab-name1.  
ENDLOOP.  
  
  
---------------------------------------------------------------------------------------------------  
  
  
\*&---------------------------------------------------------------------\*  
\*& Report ZSWAP\_81\_STRS  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
REPORT ZSWAP\_81\_STRS.  
  
PARAMETERS p\_str(20) LOWER CASE.  
  
WRITE p\_str.  
  
ULINE.  
PARAMETERS p\_str1(20).  
DATA lv\_new(30).  
CONCATENATE p\_str p\_str1 INTO lv\_new.  
WRITE lv\_new.  
ULINE.  
WRITE lv\_new+3(1)." p is printed .  
WRITE lv\_new+3. "upto 3 char's printed.  
ULINE.  
CONDENSE lv\_new.  
WRITE :/ 'condensed string without extra gaps',lv\_new.  
CONDENSE lv\_new NO-GAPS.  
WRITE :/ 'condense with no gaps',lv\_new.  
ULINE.  
DO.  
  IF lv\_new CA 's'.  
    REPLACE 's' with 'Replaced' INTO lv\_new.  
  ELSE.  
    ENDIF.  
ENDDO.  
WRITE lv\_new .  
  
  
  
---------------------------------------------------------------------------------------------------