\*&---------------------------------------------------------------------\*  
\*& Report ZSWAP\_ITAB\_PROGS  
\*&---------------------------------------------------------------------\*  
\*&  
\*&---------------------------------------------------------------------\*  
report ZSWAP\_ITAB\_PROGS.  
  
\*DATA: BEGIN OF line,  
\*      col\_1 type  C,  
\*      col\_2(4) type C,  
\*     END OF line.  
\*  
\*  
\*DATA itab LIKE TABLE OF line.  
\*DATA wa\_tab like table of itab.  
\*  
\*line-col\_1 = 'hello'.  
\*line-col\_2 = 'swap3333'.  
\*append line to itab.  
\*  
\*\*CLEAR itab.  
\*\*REFRESH itab.  
\*If itab is INITIAL.  
\*  write : 'interanl tab is empty '.  
\*  else.  
\*\*    write : 'internal tab is not mty'.  
\*    LOOP AT itab into line.  
\*      write :/ line-col\_1,"only h is printed due to size is not declared  
\*      line-col\_2. " only swap is printed due to size is 4 only  
\*    ENDLOOP.  
\*ENDIF.  
  
  
  
\*\*\*\* INSERT with sorted itab ans starndard itab.  
  
\*DATA: BEGIN OF line1,  
\*      age type I ,  
\*      name(10) type  C,  
\*      place(10) type c,  
\*     END OF line1.  
\*  
\*TYPES: BEGIN OF line,  
\*      name(10) type  C,  
\*      age type I,  
\*      place(10) type c,  
\*     END OF line.  
\*  
\*  
\*\*DATA itab like STANDARD TABLE OF line1 with NON-UNIQUE Key name age place.  
\*DATA itab like sorted TABLE OF line1 with NON-UNIQUE Key  age "name place.  
\*\*DATA itab like STANDARD TABLE OF line1 with NON-UNIQUE SORTED KEY  .  
\*  
\*  
\*line1-age = 24.  
\*line1-name = 'swapnith'.  
\*line1-place = 'hyd'.  
\*insert line1 into table itab.  
\*line1-name = 'swapnith'.  
\*line1-age = 22.  
\*line1-place = 'hyd'.  
\*insert line1 into table itab.  
\*line1-name = 'swapnith'.  
\*line1-age = 20.  
\*line1-place = 'hyd'.  
\*insert line1 into table itab.  
\*line1-name = 'swapnith'.  
\*line1-age = 24.  
\*line1-place = 'hyd'.  
\*insert line1 into table itab.  
\*  
\*LOOP AT itab INTO LINE1.  " itab to fld  
\*  WRITE: / LINE1-name,  LINE1-age,  LINE1-place.  
\*ENDLOOP.  
  
  
\* #3 INSERT LINE  
  
\*data: begin of LINE,  " fld strng  
\*        COL\_1 type I,  
\*        COL\_2 type I,  
\*      end of LINE.  
\*  
\*data ITAB like table of LINE.  
\*  
\*  
\*do 2 times.  
\*  LINE-COL\_1 = SY-INDEX.  
\*  LINE-COL\_2 = SY-INDEX \*\* 2.  
\*  append LINE to ITAB.  
\*\*  INSERT line into itab. dump error  
\*enddo.  
\*  
\*loop  at ITAB into LINE.  
\*  LINE-COL\_1 = 3 \* SY-TABIX.  
\*  LINE-COL\_2 = 5 \* SY-TABIX.  
\*  insert LINE into ITAB.  
\*  write: sy-tabix,'tabix'.  
\*endloop.  
\*loop at ITAB into LINE.  
\*  write: / SY-TABIX, LINE-COL\_1, LINE-COL\_2.  
\*endloop.  
\*  
\*DATA: BEGIN OF LINE,  
\*  col1 TYPE I,  
\*  col2 TYPE I,  
\*END OF LINE.  
\*  
\*DATA itab LIKE TABLE OF LINE  
\*  
\*  
\*DO 3 TIMES.  
\*  LINE-col1 = sy-INDEX.  
\*  LINE-col2 = sy-INDEX \*\* 2.  
\*  APPEND LINE TO itab.  
\*ENDDO.  
  
\*LOOP AT itab INTO LINE.  
\*  IF sy-tabix = 2.  
\*    LINE-col1 = sy-tabix  \*  10.  
\*    LINE-col2 = ( sy-tabix  \* 10 ) \*\* 2.  
\*    MODIFY itab FROM LINE. " modifies 2nd line  
\*\*    insert line into itab. " add new 2nd line to itab  
\*  
\*  ENDIF.  
\*ENDLOOP.  
\*line-col1 = 99."" no effect  
\*delete table itab : from line. deleting first row  
\*\*\*\*\*LOOP AT itab into LINE.  
\*\*\*\*\*  if sy-tabix = 2.  
\*\*\*\*\*     delete table itab : from line.  
\*\*\*\*\*  ENDIF.  
\*\*\*\*\*ENDLOOP." deletes first row for true now for next loop tabix is 2 again and first deleted again  
\*  
\*LOOP AT itab INTO LINE.  
\*  WRITE: / sy-tabix, LINE-col1, LINE-col2.  
\*ENDLOOP.  
  
\*DELETE  
\*\*  
\*DATA: BEGIN OF LINE,  
\*  col1 TYPE I,  
\*  col2 TYPE I,  
\*END OF LINE.  
\*DATA itab LIKE HASHED TABLE OF LINE WITH UNIQUE key col1.  
\*  
\*DO 4 TIMES.  
\*  LINE-col1 = sy-INDEX.  
\*  LINE-col2 = sy-INDEX \*\* 2.  
\*  INSERT LINE INTO TABLE itab.  
\*ENDDO.  
\*LINE-col1 = 2.  
\*DELETE TABLE itab: FROM LINE ,  
\*WITH TABLE KEY col1 = 3.  
\*  
\*\*\*\*\*\*\*only delete statement first row is deleted.  
\*\*\*\*\*\*\*with line-col=2 we can delete specifi row .  
\*\*\*\*\*\*\* with with table key also we can delete specifi row in hashed table  
\*\*" delete table itab:form line.  
\*\*  
\*LOOP AT itab INTO LINE.  
\*  WRITE: / LINE-col1, LINE-col2.  
\*ENDLOOP.  
  
  
  
\*COLLECT  
\*  
\*DATA: BEGIN OF LINE,  
\*  col\_1(3) TYPE C,  
\*  col\_2(2) TYPE n,  
\*  col\_3    TYPE I,  
\*END OF LINE.  
\*  
\*DATA itab LIKE SORTED TABLE OF LINE  
\*      WITH NON-UNIQUE KEY col\_1  col\_2.  
\*  
\*LINE-col\_1 = 'UST'.  
\*LINE-col\_2 = '12'.  
\*LINE-col\_3 = 3.  
\*  
\*COLLECT LINE INTO itab.  
\*WRITE / sy-tabix.  
\*  
\*  
\*LINE-col\_1 = 'TECH'.  
\*LINE-col\_2 = '34'.  
\*LINE-col\_3 = 5.  
\*COLLECT LINE INTO itab.  
\*WRITE / sy-tabix.  
\*  
\*  
\*LINE-col\_1 = 'UST'.  
\*LINE-col\_2 = '22'.  
\*LINE-col\_3 = 7.  
\*COLLECT LINE INTO itab.  
\*  
\*WRITE / sy-tabix.  
\*LOOP AT itab INTO LINE.  
\*  WRITE: / LINE-col\_1, LINE-col\_2, LINE-col\_3.  
\*ENDLOOP.  
  
\*Data Types  
  
\*\*--------------------------------------------------------------\*  
\*TYPES: BEGIN OF ty\_student,  
\*  ID(5)    TYPE n,  
\*  name(10) TYPE C,  
\*END OF ty\_student.  
\*  
\*  
\*\*READ + INDEX + KEY  
\*\*--------------------------------------------------------------\*  
\*\*Data Declaration  
\*\*--------------------------------------------------------------\*  
\*DATA: gwa\_student TYPE ty\_student.  
\*DATA: it TYPE TABLE OF ty\_student.  
\*  
\*gwa\_student-ID    = 1.  
\*gwa\_student-name  = 'JOHN'.  
\*APPEND gwa\_student TO it.  
\*  
\*gwa\_student-ID    = 2.  
\*gwa\_student-name  = 'JIM'.  
\*APPEND gwa\_student TO it.  
\*  
\*gwa\_student-ID    = 3.  
\*gwa\_student-name  = 'JACK'.  
\*APPEND gwa\_student TO it.  
\*  
\*SORT IT BY ID.  
\*\*\*\* # 1  
\*READ TABLE it INTO gwa\_student INDEX 1.  
\*  
\*IF sy-subrc = 0.  
\*  WRITE: gwa\_student-ID, gwa\_student-name.  
\*ELSE.  
\*  WRITE 'No Record Found'.  
\*ENDIF.  
\*  
\*\*\*LOOP AT  
\*\*\*    LOOP AT  
\*\*\*ENDLOOP.  
\*  
\*  
\*\*\*\*\*\* # 2  
\*READ TABLE it INTO gwa\_student WITH KEY name = 'JACK'.  
\*IF sy-subrc = 0.  
\*  WRITE: gwa\_student-ID, gwa\_student-name.  
\*ELSE.  
\*  WRITE 'No Record Found'.  
\*ENDIF.  
  
  
\*\*\*\*FIELD-SYMBOLS  &C  
  
\*\*\*\*\*\*\*\*\* # 1  
\*\*DATA : F1(20) TYPE C.  
\*\*  
\*\*FIELD-SYMBOLS <FS\_UST>. " serve as pointers they does hold memory for themselves  
\*\*\* instead point to referred variable.  
\*\*  
\*\*F1 = 'UST TECH'.  
\*\*  
\*\*ASSIGN F1 TO <FS\_UST>.  
\*\*  
\*\*WRITE : <FS\_UST>.  
  
\*----------------------------------  
\* # 2  
\*----------------------------------  
\*data: begin of line,  " field string  
\*        col\_1 type i,  
\*        col\_2 type i,  
\*      end of line.  
\*  
\*data itab like sorted table of line with unique key col\_1.  
\*  
\*field-symbols <fs> like line of itab.  
\*  
\*  
\*do 10 times.  
\*  line-col\_1   =   sy-index.  
\*  line-col\_2   =   2   \*   sy-index.  
\*  append line to itab.  
\*enddo.  
\*  
\*LOOP AT itab INTO LINE  .  " itab to fld  
\*  WRITE: / LINE-col\_1, LINE-col\_2.  " LINE-weight, LINE-land.  
\*ENDLOOP.  
\*  
\*\*\*\*\*\*\*\*\*\*\*\*\*\*READ TABLE<<<<<<<<<  
\*  
\*read table itab assigning <fs> index 5.  
\*\*CLEAR <fs>.  
\*write:   sy-subrc, sy-tabix.  
\*write: / <fs>-col\_1, <fs>-col\_2.