

## Internal Tables

```
**&-----*
**& Report ZAKHILA_INT_TABLES
**&-----*
**&
**&-----*

REPORT zakhila_int_tables.
TYPES :BEGIN OF ty_makt,
        matnr TYPE makt-matnr,
        spras TYPE makt-spras,
        maktx TYPE makt-maktx,
        maktg TYPE makt-maktg,
    END OF ty_makt.

DATA : it_makt TYPE TABLE OF ty_makt,
        wa_makt LIKE LINE OF it_makt.

wa_makt-matnr = '1001'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'akhila'.
APPEND wa_makt TO it_makt.

wa_makt-matnr = '1012'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'uthara'.
APPEND wa_makt TO it_makt.

wa_makt-matnr = '102'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'anu'.
APPEND wa_makt TO it_makt.

wa_makt-matnr = '103'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'akshara'.
APPEND wa_makt TO it_makt.

READ TABLE it_makt INTO wa_makt INDEX 4.
READ TABLE it_makt INTO wa_makt WITH KEY matnr = '20' spras = 'D' .

WRITE wa_makt.

READ TABLE it_makt INTO wa_makt WITH KEY matnr = '102' spras = 'E'.
WRITE wa_makt.
```

### Sorted Table

```
types :begin of ty_makt,
        matnr type makt-matnr,
        spras type makt-spras,
        maktx type makt-maktx,
        maktg type makt-maktg,
    end of ty_makt.

DATA : it_makt TYPE SORTED TABLE OF ty_makt WITH UNIQUE KEY matnr,
```

```

        wa_makt LIKE LINE OF it_makt.

wa_makt-matnr = '105'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'akhila'.
INSERT wa_makt INTO TABLE it_makt.

wa_makt-matnr = '109'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'uthara'.
INSERT wa_makt INTO TABLE it_makt.

wa_makt-matnr = '102'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'anu'.
INSERT wa_makt INTO TABLE it_makt.

wa_makt-matnr = '103'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'akshara'.
INSERT wa_makt INTO TABLE it_makt.

wa_makt-matnr = '100'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'appu'.
INSERT wa_makt INTO TABLE it_makt.

READ TABLE it_makt INTO wa_makt WITH KEY matnr = '102' BINARY SEARCH.
READ TABLE it_makt INTO wa_makt WITH KEY matnr = '103' spras = 'E'.
LOOP AT it_makt INTO wa_makt.
    WRITE :wa_makt.
ENDLOOP.

```

#### Hashed Table

```

data : it_makt type hashed table of ty_makt with unique key matnr,
      wa_makt like line of it_makt.

wa_makt-matnr = '105'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'akhila'.
INSERT wa_makt INTO TABLE it_makt.

wa_makt-matnr = '109'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'uthara'.
INSERT wa_makt INTO TABLE it_makt.

wa_makt-matnr = '102'.
wa_makt-spras = 'E'.
wa_makt-maktx = 'anu'.
INSERT wa_makt INTO TABLE it_makt.
READ TABLE it_makt INTO wa_makt WITH TABLE KEY matnr = '102'.
*sort it_makt by matnr.
SORT it_makt BY maktx.
LOOP AT it_makt INTO wa_makt.

```

```
WRITE wa_makt.  
ENDLOOP.
```

---

## Sales using Internal Tables

```
-----*
```

```
REPORT ZAISH_PURCHASE_INTERNAL_TABLE.
```

  

```
TYPES: BEGIN OF ty_vbap,  
      vbeln TYPE vbap-vbeln,  
      posnr TYPE vbap-posnr,  
      pstyv TYPE vbap-pstyv,  
      arktx TYPE vbap-arktx,  
      kwmeng TYPE vbap-kwmeng,  
      zieme TYPE vbap-zieme,  
      netwr TYPE vbap-netwr,  
      END OF ty_vbap.  
TYPES :BEGIN OF ty_vbak,  
      vbeln TYPE vbak-vbeln,  
      audat TYPE vbak-audat,  
      netwr TYPE vbak-netwr,  
      kunnr TYPE vbak-kunnr,  
      END OF ty_vbak.  
TYPES : BEGIN OF ty_t006a,  
      msehi TYPE t006a-msehi,  
      msehl TYPE t006a-msehl,  
      END OF ty_t006a.  
TYPES : BEGIN OF ty_knal ,  
      kunnr TYPE knal-kunnr,  
      namel TYPE knal-namel,  
      END OF ty_knal.
```

  

```
DATA : it_vbap TYPE table of ty_vbap,  
      wa_vbap LIKE LINE OF it_vbap,  
      it_vbak TYPE table of ty_vbak,  
      wa_vbak LIKE LINE OF it_vbak,  
      it_t006a TYPE TABLE OF ty_t006a,  
      wa_t006a like line of it_t006a,  
      it_knal TYPE TABLE OF ty_knal,  
      wa_knal LIKE LINE OF it_knal,  
      prevbeln TYPE vbap-vbeln,  
      net_sum TYPE vbap-kwmeng,  
      order_sum TYPE vbap-kwmeng.
```

  

```
SELECT-OPTIONS s_num FOR wa_vbak-vbeln.  
SELECT vbeln audat netwr kunnr FROM vbak INTO TABLE it_vbak WHERE vbeln IN  
s_num.  
SELECT kunnr namel FROM knal INTO TABLE it_knal FOR ALL ENTRIES IN it_vbak  
WHERE kunnr EQ it_vbak-kunnr.  
SELECT vbeln posnr pstyv arktx kwmeng zieme netwr FROM vbap INTO TABLE  
it_vbap WHERE vbeln in s_num.
```

```
SELECT msehi msehl FROM t006a INTO TABLE it_t006a FOR ALL ENTRIES IN it_vbap
where spras EQ sy-langu AND msehi EQ it_vbap-zieme.
```

```
LOOP AT it_vbap INTO wa_vbap.
```

```
At new vbeln.
```

```
READ TABLE it_vbak INTO wa_vbak WITH KEY vbeln = wa_vbap-vbeln.
```

```
READ TABLE it_kna1 INTO wa_kna1 WITH KEY kunnr = wa_vbak-kunnr.
```

```
WRITE :/ ' ',
       / 'Document Number:',wa_vbak-vbeln,
       / 'Doc Date:',wa_vbak-audat,
       / 'Net Val:',wa_vbak-netwr,
       / 'Sold To Party:',wa_kna1-name1,
       / 'Item',
       20 'Material',
       35 'ItemDesc',
       60 'Order Qty' ,
       80 'UoM',
       100'NetVal'.
```

```
ENDAT.
```

```
READ TABLE it_t006a INTO wa_t006a WITH KEY msehi = wa_vbap-zieme.
```

```
WRITE : / wa_vbap-posnr,
       20 wa_vbap-pstyv,
       35 wa_vbap-arktx,
       50 wa_vbap-kwmeng,
       80 wa_t006a-msehl,
       90 wa_vbap-netwr.
```

```
AT END OF vbeln.
```

```
SUM.
```

```
WRITE : /50 wa_vbap-kwmeng,
       90 wa_vbap-netwr.
```

```
ENDAT.
```

```
ENDLOOP.
```

---

## Sales using - SUBROUTINE .

---

```
REPORT ZAISH_PURCHASE_SUBROUTINE.
```

### Main Program

```
-----
```

```
TYPES: BEGIN OF ty_vbap,
       vbeln type vbap-vbeln,
       posnr TYPE vbap-posnr,
       pstyv TYPE vbap-pstyv,
       arktx TYPE vbap-arktx,
       kwmeng TYPE vbap-kwmeng,
       zieme TYPE vbap-zieme,
       netwr TYPE vbap-netwr,
       END OF ty_vbap.
TYPES :BEGIN OF ty_vbak,
```

```

        vbeln TYPE vbak-vbeln,
        audat TYPE vbak-audat,
        netwr TYPE vbak-netwr,
        kunnr type vbak-kunnr,
END OF ty_vbak.
TYPES : BEGIN OF ty_t006a,
        msehi type t006a-msehi,
        msehl type t006a-msehl,
END OF ty_t006a.
TYPES : BEGIN OF ty_knal ,
        kunnr type knal-kunnr,
        namel type knal-namel,
END OF ty_knal.
DATA : it_vbap TYPE table of ty_vbap,
        wa_vbap LIKE LINE OF it_vbap,
        it_vbak TYPE table of ty_vbak,
        wa_vbak LIKE LINE OF it_vbak,
        it_t006a TYPE TABLE OF ty_t006a,
        wa_t006a like line of it_t006a,
        it_knal TYPE TABLE OF ty_knal,
        wa_knal LIKE LINE OF it_knal,
        prevbeln TYPE vbap-vbeln,
        sum_order type vbap-kwmeng,
        sum_net type vbap-netwr,
        c_vbeln type vbak-vbeln.

TYPES : ty_it_vbak TYPE STANDARD TABLE OF ty_vbak,
        ty_it_knal TYPE STANDARD TABLE OF ty_knal,
        ty_it_vbap TYPE STANDARD TABLE OF ty_vbap,
        ty_it_t006a TYPE STANDARD TABLE OF ty_t006a.
SELECT-OPTIONS s_num FOR wa_vbak-vbeln.

START-OF-SELECTION.

PERFORM header IN PROGRAM ZAISH_PURCHASE_LIST USING s_num[] CHANGING
it_vbak it_knal.

END-OF-SELECTION.
        LOOP AT it_vbak INTO wa_vbak.
READ TABLE it_knal INTO wa_knal WITH KEY kunnr = wa_vbak-kunnr.
AT FIRST.
        WRITE :/20 'Document Number',
        40 'Doc Date',
        70 'Net Val',
        90 'Sold To Party'.
ENDAT.

        WRITE : /20 wa_vbak-vbeln,
        40 wa_vbak-audat,
        60 wa_vbak-netwr,
        90 wa_knal-namel.
*      HIDE: wa_vbak-vbeln.
        ENDLOOP.
AT LINE-SELECTION.

```

```

DATA : fnam(20),fval(50).
DATA v_vbeln TYPE vbap-vbeln.
GET CURSOR FIELD fnam VALUE fval.
fnam = wa_vbak-vbeln.
CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
  EXPORTING
    input      = fval
IMPORTING
  OUTPUT      = v_vbeln.

PERFORM items IN PROGRAM ZAISH_PURCHASE_LIST USING v_vbeln CHANGING
it_vbap it_t006a.

LOOP AT it_vbap into wa_vbap.
  READ TABLE it_t006a INTO wa_t006a WITH KEY msehi = wa_vbap-zieme.
  AT FIRST.
    WRITE :/ 'Item',
           20 'Material',
           35 'ItemDesc',
    60 'Order Qty' ,
    80 'UoM',
    100'NetVal'.
    ENDAT.
    WRITE : / wa_vbap-posnr,
           20 wa_vbap-pstyv,
           35 wa_vbap-arktx,
           50 wa_vbap-kwmeng,
           80 wa_t006a-msehl,
           90 wa_vbap-netwr.

  ENDLOOP.

```

---

```

REPORT ZAISH_PURCHASE_LIST.

```

**Sub prgm**

---

```

TYPES : BEGIN OF ty_range,
  sign type c length 1,
  opti type c length 2,
  low type vbeln,
  high type vbeln,
  END OF ty_range.
TYPES: BEGIN OF ty_vbap,
  vbeln type vbap-vbeln,
  posnr TYPE vbap-posnr,
  pstyv TYPE vbap-pstyv,
  arktx TYPE vbap-arktx,
  kwmeng TYPE vbap-kwmeng,
  zieme TYPE vbap-zieme,
  netwr TYPE vbap-netwr,
  END OF ty_vbap.
TYPES :BEGIN OF ty_vbak,
  vbeln TYPE vbak-vbeln,

```

```

        audat TYPE vbak-audat,
        netwr TYPE vbak-netwr,
        kunnr type vbak-kunnr,
END OF ty_vbak.
TYPES : BEGIN OF ty_t006a,
        msehi type t006a-msehi,
        msehl type t006a-msehl,
END OF ty_t006a.
TYPES : BEGIN OF ty_kna1 ,
        kunnr type kna1-kunnr,
        name1 type kna1-name1,
END OF ty_kna1.
DATA :
        it_vbak TYPE table of ty_vbak,
        wa_vbak LIKE LINE OF it_vbak.

TYPES : ty_it_range TYPE STANDARD TABLE OF ty_range.

TYPES : ty_it_vbak TYPE STANDARD TABLE OF ty_vbak,
ty_it_kna1 TYPE STANDARD TABLE OF ty_kna1,
ty_it_vbap TYPE STANDARD TABLE OF ty_vbap,
ty_it_t006a TYPE STANDARD TABLE OF ty_t006a.
*ty_s_num TYPE STANDARD TABLE OF vbak-vbeln.
*
*SELECT-OPTIONS s_num FOR wa_vbak-vbeln.
*&-----*
*& Form header
*&-----*
*& text
*&-----*
*&      --> S_NUM
*&      <-- IT_VBAK
*&      <-- IT_KNA1
*&-----*
FORM header  USING      p_s_num TYPE ty_it_range
                CHANGING p_it_vbak TYPE ty_it_vbak
                p_it_kna1 TYPE ty_it_kna1.
SELECT  vbeln audat netwr kunnr FROM vbak INTO TABLE p_it_vbak WHERE vbeln
IN p_s_num.
SELECT  kunnr name1 FROM kna1 INTO TABLE p_it_kna1 FOR ALL ENTRIES IN
p_it_vbak WHERE kunnr EQ p_it_vbak-kunnr.

ENDFORM.
*&-----*
*& Form items
*&-----*
*& text
*&-----*
*&      --> V_VBELN
*&      <-- IT_VBAP
*&      <-- IT_T006A
*&-----*
FORM items  USING      p_v_vbeln TYPE vbap-vbeln
                CHANGING p_it_vbap TYPE ty_it_vbap

```

```

        p_it_t006a TYPE ty_it_t006a.
SELECT  vbeln posnr pstyv arktx kwmeng zieme netwr FROM vbap INTO TABLE
p_it_vbap WHERE vbeln EQ p_v_vbeln.
SELECT msehi msehl FROM t006a INTO TABLE p_it_t006a FOR ALL ENTRIES IN
p_it_vbap where spras Eq sy-langu AND msehi EQ p_it_vbap-zieme.

ENDFORM.

```

---

## Method 2

```

*&-----*
*& Using Header & Item Tables
*&-----*
*& Main prgm
*&-----*
REPORT zaish_purchase_subroutine.

*DATA :it_vbap  TYPE zaish_vbap_table,
*      wa_vbap  LIKE LINE OF it_vbap,
*      it_t006a TYPE zaish_t006a_table,
*      wa_t006a LIKE LINE OF it_t006a.

DATA :it_header TYPE ZAISH_HEADER_TABLE,
      wa_header LIKE LINE OF it_header,
      it_item   TYPE ZAISH_ITEM_TABLE,
      wa_item   LIKE LINE OF it_item.

SELECT-OPTIONS s_num FOR wa_header-vbeln.

START-OF-SELECTION.

    PERFORM display IN PROGRAM zaish_purchase_list USING s_num[] CHANGING
it_header.
    PERFORM display_header.

END-OF-SELECTION.

AT LINE-SELECTION.
    DATA : fnam(20), fval(50).
    DATA v_vbeln TYPE vbap-vbeln.
    GET CURSOR FIELD fnam VALUE fval.

    PERFORM conversion.

clear it_item.

    PERFORM itemdetails IN PROGRAM zaish_purchase_list USING v_vbeln CHANGING
it_item.
    PERFORM display_item.

*&-----*

```



```

*& Form display_header
*&-----*
*& text
*&-----*
*& --> p1      text
*& <-- p2      text
*&-----*
FORM display_header .
  WRITE :/20 'Document Number',
        40 'Doc Date',
        70 'Net Val',
        90 'Sold To Party'.

  LOOP AT it_header INTO wa_header.
    WRITE : /20 wa_header-vbeln,
            40 wa_header-audat,
            70 wa_header-netwr LEFT-JUSTIFIED,
            90 wa_header-nam1.
  ENDLOOP.
ENDFORM.
*&-----*
*& Form display_item
*&-----*
*& text
*&-----*
*& --> p1      text
*& <-- p2      text
*&-----*
FORM display_item .
  LOOP AT it_item INTO wa_item.
    AT FIRST.
      WRITE :/ 'Item',
              20 'Material',
              35 'ItemDesc',
              60 'Order Qty' ,
              80 'UoM',
              100'NetVal'.
    ENDAT.
    WRITE : / wa_item-posnr,
            20 wa_item-pstyv,
            35 wa_item-arktx,
            50 wa_item-kwmeng,
            80 wa_item-msehi,
            90 wa_item-netwr.
  AT END OF vbeln.
    SUM.
    WRITE : /50 wa_ITEM-kwmeng,
            90 wa_ITEM-netwr.
    ENDAT.
  ENDLOOP.

ENDFORM.
*&-----*
*& Form conversion
*&-----*
*& text
*&-----*

```

```

*& --> p1          text
*& <-- p2          text
*&-----*
FORM conversion .

    CALL FUNCTION 'CONVERSION_EXIT_ALPHA_INPUT'
      EXPORTING
        input  = fval
      IMPORTING
        output = v_vbeln.
ENDFORM.

```

---

```

*&-----*
*& Report ZAISH_PURCHASE_LIST
*&-----*
*& Subroutine prgm
*&-----*
REPORT zaish_purchase_list.

TYPES : BEGIN OF ty_range,
        sign TYPE c LENGTH 1,
        opti TYPE c LENGTH 2,
        low  TYPE vbeln,
        high TYPE vbeln,
      END OF ty_range.

TYPES: ty_it_range TYPE STANDARD TABLE OF ty_range.

FORM display USING p_s_num TYPE ty_it_range
              CHANGING p_it_header TYPE ZAISH_HEADER_TABLE.

DATA : it_vbak TYPE STANDARD TABLE OF zaish_vbak,
      wa_vbak LIKE LINE OF it_vbak,
      it_knal TYPE STANDARD TABLE OF zaish_knal,
      wa_knal LIKE LINE OF it_knal.

DATA wa_header LIKE LINE OF p_it_header.

SELECT vbeln audat netwr kunnr FROM vbak INTO TABLE it_vbak WHERE vbeln IN
p_s_num.
SELECT kunnr namel FROM knal INTO TABLE it_knal FOR ALL ENTRIES IN it_vbak
WHERE kunnr EQ it_vbak-kunnr.

LOOP AT it_vbak INTO wa_vbak.
  READ TABLE it_knal INTO wa_knal WITH KEY kunnr = wa_vbak-kunnr.
  MOVE-CORRESPONDING wa_vbak TO wa_header.
  MOVE-CORRESPONDING wa_knal TO wa_header.
  APPEND wa_header TO p_it_header.

ENDLOOP.

ENDFORM.

FORM itemdetails USING v_vbeln TYPE vbap-vbeln

```

```
      CHANGING p_it_item TYPE ZAISH_ITEM_TABLE.

DATA: it_vbap  TYPE STANDARD TABLE OF zaish_vbap,
      wa_vbap  LIKE LINE OF it_vbap,
      it_t006a TYPE STANDARD TABLE OF zaish_t006a,
      wa_t006a LIKE LINE OF it_t006a.

DATA wa_item LIKE LINE OF p_it_item.

SELECT vbeln posnr pstyv arktx kwmeng zieme netwr FROM vbap INTO TABLE
it_vbap WHERE vbeln EQ v_vbeln.
SELECT msehi msehl FROM t006a INTO TABLE it_t006a FOR ALL ENTRIES IN
it_vbap WHERE spras EQ sy-langu AND msehi EQ it_vbap-zieme.

LOOP AT it_vbap INTO wa_vbap.
  READ TABLE it_t006a INTO wa_t006a WITH KEY msehi = wa_vbap-zieme.

  MOVE-CORRESPONDING wa_vbap TO wa_item.
  MOVE-CORRESPONDING wa_t006a TO wa_item.
  APPEND wa_item TO p_it_item.
ENDLOOP.
ENDFORM.
```

---

---