**INTERNAL TABLES**

\*   It is temporary  memory  location , which is used to store the data of database.

\*   Internal table can store multiple records. i.e; 8KB size is allocated for internal   tables.

\*   By default we cannot do any operation on internal tables.

\*   If we want to do any operation on internal tables, then we have to create a separate    memory called as  " WORK AREA".

WORK AERA:

It is also a temporary memory location , which can store a single record in the memory.

We can do any operation on the work areas.

Once the operation is finished on work area, then we modify the internal Tables.

Syntax for Internal  tables:

|  |
| --- |
| DATA<ITABNAME> TYPE  TABLE  OF  <DB TABLE NAME>  DATA<ITABNAME>  TYPE  TABLE  OF  <USER-DEFINED TABLE>  DATA<ITABNAME>  TYPE  TABLE  OF  <DB.TABLE> OCCURS 0 WITH         HEADER   LINE. |

Syntax for  Work Area:

|  |
| --- |
| DATA <WA\_NAME>  TYPE  <DB.TABLE>  DATA <WA\_NAME>  TYPE  <USER-DEFINED TYPE> |

SELECT:

It is  a  keyword, which is used  to select  the data from DataBase  table into Internal Table (or)WorkArea.

Syntax:

|  |
| --- |
| SELECT   \*  (or) F1  F2  F3 - - - -                        From <DB.Table>                        Into   <ITab/ WA>                        Where   F1  = <Value>                        And       F2  = <Value>                                        .                                        . |

LOOP.........ENDLOOP:

It  is a Keyword, which is used to  LOOP  (or)  read each record from Internal Table to  Work Area.

Syntax:

|  |
| --- |
| LOOP AT <ITAB>  INTO <WA>    ---    ---    ---  ENDLOOP. |

Ex  1:

Report  ZEX\_PRG.

\* Data Declarations

DATA:  it\_ZCUST\_TABLE TYPE OF  ZCUST\_TABLE.

DATA:  WA\_ZCUST\_TABLE TYPE ZCUST\_TABLE.

Select  \*  from  ZCUST\_TABLE  into Table  it\_ZCUST\_TABLE.

Loop At it\_ZCUST\_TABLE  into  WA\_ZCUST\_TABLE.

      Write : /  WA\_ZCUST\_TABLE-- CUSTNO,

                      WA\_ZCUST\_TABLE-- CNAME,

                        WA\_ZCUST\_TABLE-- CITY,

                        WA\_ZCUST\_TABLE-- GENDER.

 Endloop.

Ex 2:

Report   ZKNA1\_PRG.

DATA : it\_KNA1  TYPE  TABLE OF   KNA1.

DATA : WA\_KNA1  TYPE  KNA1.

Select  \*  from  KNA1  into  it\_KNA1.

Loop at  it\_KNA1  into WA\_KNA1.

    write : /  WA\_KNA1-- KUNNR,

                    WA\_KNA1-- NAME1,

                    WA\_KNA1-- LAND1,

                     WA\_KNA1-- ORT01.

End loop.

USER-DEFINED TYPE:

Types  is a keyword, which is used to defined User-defined fields for internal table (or) WorkArea instead of all the fields from  "DataBase Tables".

Syntax:

|  |
| --- |
| Types : Begin Of <TypeName>,                 F1 (Length)  Type  <DataType>,                 F2                   Type  <TableName-Fname>,                 F3                   Type  <TableName--Fname>,                  .                                        .                  .                                        .                Endof  <TypeName>. |

Ex 1:

Report  ZType-Table.

Type: Begin of  TY\_MARA,

            MATNR(18)  type  MARA--MATNR,

            MTART          type  MARA--MTART,

            MBRSH          type  MARA--MTART,

            End of TY\_MARA.

DATA : I\_TY\_MARA  type table of  TY\_MARA.

DATA : WA\_TY\_MARA  type TY\_MARA.

Select  MATNR  MTART  MBRSH  from MARA into  table I\_TY\_MARA.

LOOP AT  I\_TY\_MARA  into  WA\_TY\_MARA.

    write :/  WA\_TY\_MARA--MATNR,

                   WA\_TY\_MARA--MTART,

                   WA\_TY\_MARA--MBRSH.

 ENDLOOP.

**Operations on  Internal  Tables:**

We have  16 different operations. They are,

1. APPEND.

2. INSERT.

3. SORT.

4. DESCRIBE TABLE.

5. READ TABLE--------- 1. WITH INDEX

                                           2. WITH KEY

6. LOOP...........ENDLOOP.

7. MODIFY.

8. DELETE.

9. DELETE ADJACENT DUPLICATES.

10. CLEAR.

11. REFRESH.

12. FREE.

13. APPEND  LINES OF

14. INSERT  LINES OF

15. MOVE ITAB1 TO  ITAB2.

16. COLLECT.

1. APPEND:

It is used to add  a single record  from WORKAREA  to INTERNAL TABLE.

The record is always added at bottom.

Syntax:

|  |
| --- |
| APPEND  <WA>  TO  <ITAB>. |

2.INSERT:

It is used to insert a record from  WA  into  ITAB  at a specified location.

Syntax:

|  |
| --- |
| INSERT <WA>  INTO <ITAB> INDEX <INDEX NO> |

3.SORT:

It is used to sort the data of  internal Table in  Ascending  (or) Descending Order.

By default it is Ascending  Order.

Syntax:

|  |
| --- |
| SORT  <ITAB>  BY  F1  F2  F3 ............. <ASCENDING/DESCENDING>. |

4.DESCRIBE TABLE:

It is  used  to find the total no. of  records in an internal table.

Syntax:

|  |
| --- |
| DESCRIBE TABLE  <ITAB1>  LINES <VARIABLE>. |

5. i. READ TABLE  WITH INDEX:

It is used to read  a single record from ITAB  into  WA  specified by  index no.

Syntax:

|  |
| --- |
| READ TABLE <ITAB>  INTO <WA> INDEX <INDEX NO>. |

 ii. READ TABLE  WITH  KEY:

It is also used to read a single record from  ITAB  into WA specified by field name and  field value.

Syntax:

|  |
| --- |
| READ TABLE <ITAB>  INTO <WA> WITH KEY  F1 name = <FVal>                                                                                              F2name = <FVal>                                                                                              F3name = <FVal>                                                                                                   .                  .                                                                                                   .                  .                                                                                              Binary search. |

NOTE:

The prerequisite for binary search is, the ITAB should be sorted based on Search criteria (or) Searching fields.

6.MODIFY:

This  statement is used to modify single  (or) multiple records based on condition.

Syntax:

|  |
| --- |
| MODIFY<ITAB> FROM <WA> INDEX <INDEX NO>         TRANSPORTING  F1 , F2 , F3 ........ |

|  |
| --- |
| MODIFY<ITAB> FROM <WA> TRANSPORTING  F1 , F2 , F3..........                                                              WHERE   F1  =  VALUE                                                              AND         F2 =  VALUE                                                                                   .                                                                                   . |

SY-TABIX:

It is a system variable which  stores the index no of   ITAB   record which is currently processed in the  WA.

TRANSPORTING:

It is a keyword which specifies the no of fields are modified from work area  to ITAB  instead of all the fields.

7.DELETE:

This statement is used to  DELETE  single (or) multiple records based on condition.

Syntax:

|  |
| --- |
| DELETE <ITAB> INDEX <INDEX NO>. |

DELETE <ITAB> WHERE  F1  =  FVal

                                 AND       F2  =  Fval

                                 AND       F3  =  Fval

                                                   .           .

8.DELETE  ADJACENT DUPLICATES:

This statement is used to delete adjacent records from the   ITAB.

Syntax:

|  |
| --- |
| DELETE ADJACENT DUPLICATES  FROM <ITAB>                                          COMPARING  F1  F2  F3  ALL  FIELDS. |

9.CLEAR:

This  statement is used to DELETE  the data from WA.

Syntax:

|  |
| --- |
| CLEAR <WA>. |

10.REFRESH:

This statement is used to DELETE the data from ITAB.

Syntax:

|  |
| --- |
| REFRESH <ITAB>. |

11.FREE:

This  statement is used to DELETE the data from  WA and ITAB.

Syntax:

|  |
| --- |
| FREE <WA/ITAB>. |

NOTE:

Diff  b/w CLEAR , REFRESH  and  FREE  is,

CLEAR , REFRESH  will delete the data but not memory where as the FREE  statement will delete the data as well as memory also.

12.APPEND LINES OF:

This statement is used to append the data from one  ITAB into another ITAB based on the selection.

The data will append at Bottom in the 2nd ITAB.

Syntax:

|  |
| --- |
| APPEND LINES OF <ITAB1> FROM <INDEX NO1>                                                              TO <INDEX NO2>                                                         INTO <ITAB2>. |

13.INSERT LINES OF:

This statement is used to insert the data from one ITAB to another ITAB at specified location based on the selection.

Syntax:

|  |
| --- |
| INSERTLINES OF <ITAB1> FROM <INDEX NO1>                                                            TO  <INDEX  NO2>                                                        INTO <ITAB2> INDEX <INDEX NO>. |

14. MOVE  ITAB1[ ] TO ITAB2[ ] :

This statement is used to move the entire data from one ITAB1 to another ITAB2.

Syntax:

|  |
| --- |
| ITAB1[ ]  =  ITAB2[ ]. |

15. COLLECT :

This statement checks weather the WA record already exists with the same key.

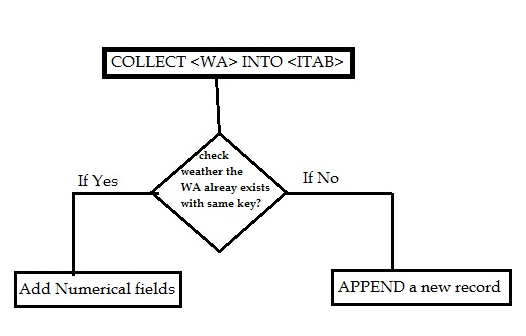
If yes, it just add numerical fields.

If no , it will append the new record.

Same key means , Same Character Fields. i.e; C , N , D , T.

Numerical fields means  I , F  , P.

Syntax:



TYPES  OF  TABLES :

      1. Transparent Tables.

      2. Pool Tables.

      3. Cluster Tables.

Comparison of Transparent, Pool and Cluster tables:

|  |  |  |
| --- | --- | --- |
| Transparent | Pool | Cluster |
| Contain a single table. Used to store master data | They are used to hold a large number of very small tables(stores system data) | They are used to hold data from a few number of large tables.(stores system data) |
| It has a one-to-one relationship with a table in the database | It has a many-to-one relationship with a table in the database | It has a many-to-one relationship with table in the database |
| For each transparent table there is one associated table in the   database | It is stored with other pooled tables in a single table called table pool in the database | Many cluster tables are stored in a single table in the database called a table cluster |
| The database table has the same name, same number of fields and the fields have the same names | The database table has different name, different number of fields and fields have different names | The database table has different name, different number of fields and fields have different names |
| There is only a single table | Table pools contain more tables than table clusters | Contains less tables than table pools |
| Secondary indexes can be created | Secondary indexes cannot be created | Secondary indexes cannot be created |

TYPES  OF  INTERNAL TABLES :

There are three types of Internaltables.

1. STANDARD internal Tables.

2. SORTED internal Tables.

3. HASHED internal Tables.

|  |  |  |
| --- | --- | --- |
| STANDARD | SORTED | HASHED |
| These are the default internal tables which are created by us.  We use either key operation (or) index operation to read a record.  We use either linear search (or)   Binary search for reading record.  If use Binary Search, the response time will be  Resp.Time  =  Log(N).  We can append , insert the records whenever we want.  We can sort the data based on our own conditions. | These are special ITAB's where the data is automatically sorted when ever a new record is added.  We use either key(or) index operation to read a record.  We use only Binary search for reading a record, bcoz the data is automatically sorted.  The Response  Time will be same.  The main disadvantage is , we can not sort ITABS based on our conditions, bcoz the data is already sorted. | These are also special type of   ITAB which should be used when working with large data sets(Bulk amount of data).  Here, we use only Keyoperation, but not the index  operation.  It uses Hashed algorithm for reading a record.  The Resp.Time is always fixed regardless of the total no of records.  In Real-Time , we hashed ITAB's only whenever we work with server to server communication like transferring the data from ABAP to BI server. |