**PL/SQL Tables** **aka Associative Arrays**

**Features of PL/SQL tables are as follows –**

1. It is a composite data type.
2. They are modeled as similar to database tables, but they are not permanent tables. So they can be created and manipulated only in a PL SQL block.
3. They can have only one column but any data type, scalar or composite.
4. It will have a primary key which is compulsory for the reference of values
5. There is no name to the column and primary key
6. The data type of the primary key is BINARY\_INTEGER.

BINARY\_INTEGER is a special data type which can be given only to the column of PL/SQL table for it’s indexing purpose to store and retrieve values.

Range of binary\_integer is -2147483647 to +2147483647

1. Size is unconstrained (Table size grows as the rows are added to the table).
2. Can visualize a PL/SQL table as a **single dimensional vertical array**, which can hold unlimited elements.
3. **Suitable for storing and displaying the values of one column of a table given by a cursor.**

***Example of PL SQL Table –***

**Each name from the emp table is given to the vname plsql table by using cursor. Then those names from vname table are displayed .**

Declare

Type nametable IS TABLE OF **CHAR(10)** INDEX BY BINARY\_INTEGER;

/\*Creating variable vname of nametable type.\*/

vname nametable;

Cursor cf is select **ename** from emp;

i number;

vrows number;

/\*i is for the loop and vrows is for displaying the total names from the vname table\*/

Begin

Open cf;

i := 1;

Loop

Fetch cf into vname(i); /\*Transferring each ename into vname table\*/

Exit when cf%NotFound;

vrows := cf%rowcount;

i := i + 1;

End Loop;

Close cf;

/\*Now retrieving the names from the vname plsql table using for loop.\*/

For n in 1..vrows

Loop

dbms\_output.put\_line('Name is '||vname(n));

End Loop;

End;

Properties of a PL SQL table ---

* Exists
* Count
* First
* Last
* Next
* Prior
* Delete

declare

Type discount is TABLE OF number

INDEX By Binary\_Integer;

d discount;

Begin

d(5) := 90;

d(2) := 50;

d(8) := 70;

d(11) := 67;

d(14) := 68;

d(1) := 1;

d(23) := 5;

d(23) := 51;

dbms\_output.put\_line('The value at 23 index number is ' || d(23));

dbms\_output.put\_line('The value at index number 6 is ' || d(6));

if **d.EXISTS(6)** Then

dbms\_output.put\_line(d(6));

else

dbms\_output.put\_line('There is no element in the sixth row');

end if;

dbms\_output.put\_line('The total number of elements in d are '|| d.count);

/\*dbms\_output.put\_line('The first index number is ' || d.FIRST);

dbms\_output.put\_line('The last index number is ' || d.LAST);

dbms\_output.put\_line('The index number after 2 is ' || d.next(2));

dbms\_output.put\_line('The index number before 8 is ' || d.prior(8));

d.delete(5);

dbms\_output.put\_line('The total number of elements in d are '|| d.count);

d.delete(11,14);

dbms\_output.put\_line('The total number of elements in d are '|| d.count);

d.delete;

dbms\_output.put\_line('The total number of elements in d are '|| d.count); \*/

end;

**Example of Delete Property:**

|  |
| --- |
| Declare  Type List is Table of Int Index by Binary\_Integer;  V List;  Begin  **For x in 1 .. 20**  **Loop**  **V(x) := x;**  **End loop;**  **dbms\_output.put\_line('Total Value Count ' || V.Count);**  **V.Delete(7); -- 7th index number value gets deleted**  **dbms\_output.put\_line('\*\*\*\*\*\*\*\*\*\* ' || V.Count);**  **V.Delete(8, 12); -- Between 8 and 12**  **dbms\_output.put\_line('$$$$$$$$$$ ' || V.Count);**  **V.Delete; -- All get deleted!!!!**  **dbms\_output.put\_line('########## ' || V.Count);**  End; |

**Table made up of %Rowtype**

|  |
| --- |
| Declare  Type nametable IS TABLE OF **EMP%ROWTYPE**  INDEX BY BINARY\_INTEGER;  /\*Creating variable vname of nametable type.\*/  vname nametable;  Cursor cf is select \* from emp;  i number;  vrows number;  /\*i is for the loop and vrows is for displaying the total names from the vname table\*/  Begin  Open cf;  i := 1;  Loop  Fetch cf into vname(**i**);  **/\* Transferring each record, means all columns into vname table \*/**  Exit when cf%NotFound;  vrows := cf%rowcount;  i := i + 1;  End Loop;  Close cf;  /\*Now retrieving the **ename, sal and job** from the vname plsql table using for loop.\*/  For n in 1..vrows  Loop  dbms\_output.put\_line('Name is '||**vname(n).ENAME**);  dbms\_output.put\_line('Name is '||**vname(n).SAL**);  dbms\_output.put\_line('Name is '||**vname(n).JOB**);  dbms\_output.put\_line('xxxxxxxxxxxxxxxxxxxxxxx');  End Loop;  End; |