

Cursors in PL/SQL

A **Cursors** is a temporary work area created in the system memory when a SQL statement is executed. It is a temporary memory which is used to fetch more than one record at a time from existing table.

This temporary work area is used to store the data retrieved from the database, and manipulate this data. A cursor can hold more than one row, but can process only one row at a time. The set of rows the cursor holds is called the active set.

Why use Cursor ?

It is a temporary memory which is used to fetch more than one record at a time from existing table.

Type of cursor

Cursor are broadly divided into two types;

- Implicit cursor
- Explicit cursor

Implicit cursor

This type of cursor is perform by the system internally those cursor can be called as implicit cursor.

Explicit cursor

This type of cursor is performed by the user manually or programatically those cursor can be called as explicit cursor.

Steps to perform cursor

Steps	Syntax
Declare the cursor	open cursor_name;
Open the cursor	open cursor_name;
Fetch the record from the cursor	fetch cursor_name into variables;
Close the cursor	close cursor_name;

Declare the cursor

Syntax

```
declare
a emp %rowtype;
cursor c is select * from emp where depno=&deptno;
begin
open c;
fetch c into a;
dbms_output.put_line(a.empno || ' ' a.ename || ' ' || a.sal);
close c;
end;
```

Note: In above exaple ruturen only one record because the statements are lies within the scopes.

% found

This attribute is used for check whether the record is found or not in the cursor. It return boolean value either true or false. If the record found then it return ture otherwise false.

Syntax

```
declare
a emp %rowtype;
cursor c is select * from emp where depno=&deptno;
begin
open c;
loop fetch c into a;
if c % found then
dbms_output.put_line(a.empno || ' ' a.ename || ' ' || a.sal);
else
exit;
end if;
end loop;
close c;
end;
```

Syntax of explicit cursor

Following is the syntax to create an explicit cursor:

```
CURSOR cursor_name IS select_statement;;
```

Steps:

You must follow these steps while working with an explicit cursor.

1. Declare the cursor to initialize in the memory.
2. Open the cursor to allocate memory.
3. Fetch the cursor to retrieve data.
4. Close the cursor to release allocated memory.

1) Declare the cursor:

It defines the cursor with a name and the associated SELECT statement.

Syntax for explicit cursor declaration

```
CURSOR name IS  
SELECT statement;
```

2) Open the cursor:

It is used to allocate memory for the cursor and make it easy to fetch the rows returned by the SQL statements into it.

Syntax for cursor open:

```
OPEN cursor_name;
```

3) Fetch the cursor:

It is used to access one row at a time. You can fetch rows from the above-opened cursor as follows:

Syntax for cursor fetch:

```
FETCH cursor_name INTO variable_list;
```

4) Close the cursor:

It is used to release the allocated memory. The following syntax is used to close the above-opened cursors.

Syntax for cursor close:

```
Close cursor_name;
```

