**PRACTICAL NO 5**

**CURSORS**

When an SQL statement is processed, Oracle creates a memory area known as context area. A cursor is a pointer to this context area. It contains all information needed for processing the statement. In PL/SQL, the context area is controlled by Cursor. A cursor contains information on a select statement and the rows of data accessed by it referred to as the **active set**.

A cursor is used to referred to a program to fetch and process the rows returned by the SQL statement, one at a time. There are two types of cursors:

* Implicit Cursors
* Explicit Cursors

**Implicit Cursors**

Implicit cursors are automatically created by Oracle whenever an SQL statement is executed, when there is no explicit cursor for the statement.

Whenever a DML statement (INSERT, UPDATE and DELETE) is issued, an implicit cursor is associated with this statement.

For INSERT operations, the cursor holds the data that needs to be inserted.

For UPDATE and DELETE operations, the cursor identifies the rows that would be affected.

Orcale provides some attributes known as Implicit cursor's attributes to check the status of DML operations. Some of them are: %FOUND, %NOTFOUND, %ROWCOUNT and %ISOPEN.

1. **Create employee table,insert 5 meaningful record to it.**

**Using Implicit cursor, Update the sql of specified employee id,if that id is available in the table.**

**Using SQL%FOUND return the appropriate message**

create table employee(id number(5) primary key,first\_name varchar2(10),last\_name varchar2(10),salary number(10));

insert into employee values(&id,'&first\_name','&last\_name',&salary);

EXECUTE THIS INSERT STATEMENT 10 TIMES TO INSERT 10 RECORDS.

declare

eid number(10);

sal number(10);

begin

update employee set salary=&sal

where id=&eid;

if SQL%found then

dbms\_output.put\_line('Salary Updated');

else

dbms\_output.put\_line('id not found');

end if;

end;

/

**Explicit Cursor:**

Programmers are allowed to create named context area to execute their DML operations to get more control over it. The explicit cursor should be defined in the declaration section of the PL/SQL block, and it is created for the ‘SELECT’ statement that needs to be used in the code.

* **Declaring the cursor:**Declaring the cursor simply means to create one named context area for the ‘SELECT’ statement that is defined in the declaration part. The name of this context area is same as the cursor name.
* **Opening Cursor:**Opening the cursor will instruct the PL/SQL to allocate the memory for this cursor. It will make the cursor ready to fetch the records.
* **Fetching Data from the Cursor :** the ‘SELECT’ statement is executed and the rows fetched is stored in the allocated memory. These are now called as active sets. Each fetch statement will fetch one active set and holds the information of that particular record.
* **Closing the Cursor**Once all the record is fetched now, we need to close the cursor so that the memory allocated to this context area will be released.

**Syntax:-**

DECLARE

CURSOR <cursor\_name> IS <SELECT statement^>

<cursor\_variable declaration>

BEGIN

OPEN <cursor\_name>;

FETCH <cursor\_name> INTO <cursor\_variable>;

.

.

CLOSE <cursor\_name>;

END;

**2) create a table employee ,insert 5 meaningful records to it. Write a cursor to accept id of the employee and print its first name and last name.**

(EMPLOYEE TABLE IS ALREADY ABOVE. THIS IS COMMENTED N USED ONLY FOR REFERENCE)

--create table employee(id number(5) primary key,first\_name varchar2(10),last\_name ----varchar2(10),salary number(10));

declare

fst\_name varchar2(10);

eid varchar2(10);

cursor emp\_curs is select first\_name from employee where id=’&eid’;

begin

open emp\_curs;

loop

fetch emp\_curs into fst\_name;

exit when emp\_curs%NOTFOUND;

dbms\_output.put\_line(fst\_name||' ');

end loop;

close emp\_curs;

end;

/

**3) create a table employee,insert 5 meaningful records to it.Create a cursor to display first name and last name of employee till the last record is found by for loop.**

declare

fst\_name varchar2(10);

sal number(14,2);

counter varchar2(5);

cursor emp\_curs is select first\_name,salary from employee;

begin

for e\_curs in emp\_curs

loop

dbms\_output.put\_line(e\_curs.first\_name||' '||e\_curs.salary); counter:=to\_char(emp\_curs%rowcount);

end loop;

dbms\_output.put\_line('no of row processed='||counter);

end;

/