Exp No-15
Creation of Cursor

AIM Create a PL/SQL Cursors

Cursors

A cursor is a named control structure used by an application program to point to and select a row of data from a result set. Instead of executing a query all at once, you can use a cursor to read and process the query result set one row at a time.

Implicit Cursors:

An implicit cursor is a session cursor that is constructed and managed by PL/SQL. PL/SQL opens an implicit cursor every time you run a SELECT or DML statement. You cannot control an implicit cursor, but you can get information from its attributes.

Explicit Cursors:

Explicit Cursors are Created by Users whenever the user requires them. Explicit Cursors are used for Fetching data from Table in Row-By-Row Manner.

How to create Explicit Cursor:

Declare Cursor Object.

Syntax : DECLARE cursor_name CURSOR FOR SELECT * FROM table_name

DECLARE s1 CURSOR FOR SELECT * FROM studDetails

Open Cursor Connection.

Syntax : OPEN cursor_connection

OPEN s1

Fetch Data from cursor.

There are total 6 methods to access data from cursor. They are as follows :

FIRST is used to fetch only the first row from cursor table.

LAST is used to fetch only last row from cursor table.

NEXT is used to fetch data in forward direction from cursor table.

PRIOR is used to fetch data in backward direction from cursor table.ABSOLUTE n is used to fetch the exact nth row from cursor table.

RELATIVE n is used to fetch the data in incremental way as well as decremental way.

Syntax : FETCH NEXT/FIRST/LAST/PRIOR/ABSOLUTE n/RELATIVE n FROM cursor_name

FETCH FIRST FROM s1

FETCH LAST FROM s1

FETCH NEXT FROM s1

FETCH PRIOR FROM s1

FETCH ABSOLUTE 7 FROM s1

FETCH RELATIVE -2 FROM s1

Close cursor connection.

Syntax : CLOSE cursor_name

CLOSE s1

Deallocate cursor memory.

Syntax : DEALLOCATE cursor_name

DEALLOCATE s1

Question 1: Consider the table

Customer (accout no, customer name,balance amount,date of join).

Write a pl/sql program using cursor to display the id, name, age, Balance, and date of join of all employees in Customer table.

```
set serveroutput on;
create table Customer(id int primary key,  name varchar(20), age int, address varchar(30), balance int, doj date);
insert into Customer values(1, 'Alice', 20, 'Address1', 150000, TO_DATE('2009/01/01','%yyyy-%mm-%dd'));
insert into Customer values(2, 'Bob', 30, 'Address2', 200000, TO_DATE('2007/01/01','%yyyy-%mm-%dd'));
insert into Customer values(3, 'Cindy', 40, 'Address3', 30000, TO_DATE('2019/01/01','%yyyy-%mm-%dd'));
insert into Customer values(4, 'Sam', 50, 'Address4', 40000, TO_DATE('2018/01/01','%yyyy-%mm-%dd'));
insert into Customer values(5, 'Eric', 60, 'Address5', 11000, TO_DATE('2008/01/01','%yyyy-%mm-%dd'));
insert into Customer values(6, 'Tom', 20, 'Address6', 60000, TO_DATE('2016/01/01','%yyyy-%mm-%dd'));
insert into Customer values(7, 'John', 30, 'Address7', 70000, TO_DATE('2015/01/01','%yyyy-%mm-%dd'));
insert into Customer values(8, 'Sari', 40, 'Address8', 80000, TO_DATE('2014/01/01','%yyyy-%mm-%dd'));
insert into Customer values(9, 'Timo', 50, 'Address9', 90000, TO_DATE('2013/01/01','%yyyy-%mm-%dd'));
insert into Customer values(10, 'Skove', 60, 'Address10', 100000,
TO_DATE('2012/01/01','%yyyy-%mm-%dd'));
select * from  Customer;


DECLARE CURSOR cursor0 IS
SELECT id, name, age, balance , doj FROM Customer;
   --variable definition
   e_name Customer.name%type;
   e_id Customer.id%type;
   e_bal Customer.balance%type;
   e_age Customer.id%type;
   e_doj Customer.doj%type;
BEGIN
   -- open cursor
   OPEN cursor0;
   LOOP
      FETCH cursor0 INTO e_id, e_name, e_age, e_bal, e_doj;
      -- exit condition
      EXIT WHEN cursor0%NOTFOUND;
```

```
    -- print employee info
    dbms_output.put_line('Customer Id:'||e_id);
    dbms_output.put_line('Customer Name:'||e_name);
    dbms_output.put_line('Customer Age:'||e_age);
    dbms_output.put_line('Balance :'||e_bal);
    dbms_output.put_line('Date of Join:'||to_char(e_doj));
    dbms_output.put_line('============================');
  END LOOP;
  -- close cursor
  CLOSE cursor0;
END;
```

Question 2:

Consider the table Customer (accout no, customer name,balance amount,date of join). Implement a PL/SQL block to insert those customers who have

current balance greater than 1 Lakh

and date of join before 1 january 2010 into the table premium customer

who doesnt meet above criteria are to be inserted into table nonpremium customer .

```
create table Premium_Customers(id int primary key, name varchar(20), age int, address
varchar(30), balance int, doj date);

create table Nonpremium_Customers(id int primary key, name varchar(20), age int, address
varchar(30), balance int, doj date);

DECLARE CURSOR cursor1 IS
SELECT * FROM Customer;
BEGIN
  -- open cursor
  FOR c1 in cursor1
    LOOP
    if c1.balance > 100000 and c1.doj  < '01-JAN-2010' then
      dbms_output.put_line('=======Premium_Customers=======');
      dbms_output.put_line('Employee Name:'||c1.name);
      insert into Premium_Customers values(c1.id, c1.name, c1.age,
            c1.address, c1.balance, c1.doj);
    else
      dbms_output.put_line('=======Nonpremium_Customers=======');
      dbms_output.put_line('Employee Name:'||c1.name);
      insert into Nonpremium_Customers values(c1.id, c1.name, c1.age,
            c1.address, c1.balance, c1.doj);
    end if;
  END LOOP;
END;

select * from Premium_Customers;
```

*select * from Nonpremium_Customers;*

Question 3:
Consider the table Account(Customer name,account number,date_last transaction,amount).

Implement a PL/SQL block to perform the following action on the table .
Calculate the interest of each person if it satisfies the condition
a)if the last transaction is not on the current month
insert the records into inactive customer
b)otherwise check the balance amount and display the interest amount
i)if the balance amount is less than 50000 interest rate is 5% of the amount
ii)if it is between 250000 and 5 Lakhs interest rate is 10%
iii)if the amount is greater than 5 lakh interest rate is 15%

```
create table AccDetails(accno number(10),cname varchar(20),lastdate date,amount number(7));

insert into AccDetails values(101,'Alice','08-JAN-2017',50000);
insert into AccDetails values(102,'Bob','10-FEB-2017',100000);
insert into AccDetails values(103,'Cindy','17-MAR-2017',25000);
insert into AccDetails values(104,'Sam','06-APR-2017',300000);
insert into AccDetails values(105,'Eric','15-MAY-2017',650000);

create table InactiveCustomer(accno number(10),cname varchar(20));

DECLARE CURSOR cursor2 IS
SELECT * FROM AccDetails;
 iRow AccDetails %rowtype;
 v_month number;
 interest number;
BEGIN
   -- open cursor
   OPEN cursor2;
   LOOP
      FETCH cursor2 INTO iRow.accno, iRow.cname, iRow.lastdate, iRow.amount;
      -- exit condition
      EXIT WHEN cursor2%NOTFOUND;
      v_month := months_between(sysdate, iRow.lastdate);

      dbms_output.put_line('iRow.cname '||iRow.cname);
      dbms_output.put_line('iRow.lastdate '||iRow.lastdate);
      dbms_output.put_line('iRow.amount '||iRow.amount);
      dbms_output.put_line('v_month '||v_month);

      if v_month>=1 then
         dbms_output.put_line('Moving to InactiveCustomer ...!');
         insert into InactiveCustomer values(iRow.accno, iRow.cname);
      else
         if iRow.amount < 250000 then
            interest := iRow.amount*0.05;
```

```
            dbms_output.put_line('interest of '||iRow.cname|| ' is :'||interest);
        elsif iRow.amount>250000 and iRow.amount<500000 then
            interest := iRow.amount*0.1;
            dbms_output.put_line('interest of '||iRow.cname|| ' is :'||interest);
        elsif iRow.amount>500000 then
            interest := iRow.amount*0.15;
            dbms_output.put_line('interest of '||iRow.cname|| ' is :'||interest);
        else
            dbms_output.put_line('Error...!');
        end if;
      end if;
   END LOOP;
   -- close cursor
   CLOSE cursor2;
END;

select * from AccDetails;
select * from InactiveCustomer;
```