EXPERIMENT: 10

Develop programs using features parameters in a CURSOR, FOR UPDATE CURSOR, WHERE CURRENT of clause and CURSOR variables.

Sol:

```
SQL> create table customers(id number(3), name varchar2(10), age number(3), address varchar2(10), salary number(10,2));
Table created.
SQL> insert into customers values(1,'ramesh',32,'ahmedabad',2000);
1 row created.
SQL> insert into customers values(2,'khilan',25,'Delhi',1500);
1 row created.
SQL> insert into customers values(3,'kaushik',23,'Kota',2000);
1 row created.
SQL> insert into customers values(4,'chitali',25,'Mumbai',6500);
1 row created.
SQL> select *from customers;
        ID NAME   AGE ADDRESS   SALARY
        1 ramesh    32 ahmedabad    2000
        2 khilan    25 Delhi    1500
        3 kaushik   23 Kota    2000
        4 chitali   25 Mumbai  6500
4 rows selected.
```

1. Using Parameterized Cursor

```
SQL> DECLARE
       CURSOR cur_customer(p_age NUMBER) IS
         SELECT id, name, salary FROM customers WHERE age = p_age;
       v_id customers.id%TYPE;
       v_name customers.name%TYPE;
       v_salary customers.salary%TYPE;
       BEGIN
            OPEN cur_customer(25);
       LOOP
         FETCH cur_customer INTO v_id, v_name, v_salary;
         EXIT WHEN cur_customer%NOTFOUND;
         DBMS_OUTPUT.PUT_LINE('ID: ' || v_id || ' Name: ' || v_name || ' Salary: ' ||v_salary);
       END LOOP;
     CLOSE cur_customer;
     END;
     /
```

ID: 2 Name: khilan Salary: 1500
ID: 4 Name: chitali Salary: 6500

PL/SQL procedure successfully completed.

2. Using FOR UPDATE CURSOR

Used when we want to update records fetched by the cursor.

```
SQL> DECLARE
       CURSOR cur_salary IS
         SELECT salary FROM customers WHERE salary < 2000 FOR UPDATE;
     BEGIN
       FOR rec IN cur_salary LOOP
         UPDATE customers SET salary = salary + 500 WHERE CURRENT OF cur_salary;
       END LOOP;
       COMMIT;
     END;
     /
```

PL/SQL procedure successfully completed.

3. Using WHERE CURRENT OF Clause

This clause helps to update/delete the row that the cursor is currently pointing to.

```
SQL> DECLARE
       CURSOR cur_customers IS
         SELECT id, salary FROM customers WHERE salary < 2000 FOR UPDATE;
       v_id customers.id%TYPE;
       v_salary customers.salary%TYPE;
     BEGIN
       OPEN cur_customers;
       LOOP
         FETCH cur_customers INTO v_id, v_salary;
         EXIT WHEN cur_customers%NOTFOUND;

         UPDATE customers SET salary = salary + 500 WHERE CURRENT OF cur_customers;
       END LOOP;
       CLOSE cur_customers;
       COMMIT;
     END;
     /
```

PL/SQL procedure successfully completed.

4. Using Cursor Variables (REF CURSOR)

A cursor variable allows for flexible data retrieval.

SQL:
```
DECLARE
   TYPE cur_type IS REF CURSOR;
   v_cursor cur_type;
   v_id customers.id%TYPE;
   v_name customers.name%TYPE;
   v_salary customers.salary%TYPE;
BEGIN
   OPEN v_cursor FOR SELECT id, name, salary FROM customers WHERE salary > 2000;

   LOOP
     FETCH v_cursor INTO v_id, v_name, v_salary;
     EXIT WHEN v_cursor%NOTFOUND;
     DBMS_OUTPUT.PUT_LINE('ID: ' || v_id || ' Name: ' || v_name || ' Salary: ' || v_salary);
   END LOOP;
   CLOSE v_cursor;
END;
/
```

ID: 4 Name: chitali Salary: 6500

PL/SQL procedure successfully completed.

SQL> select *from customers;

ID	NAME	AGE	ADDRESS	SALARY
1	ramesh	32	ahmedabad	2000
2	khilan	25	Delhi	2000
3	kaushik	23	Kota	2000
4	chitali	25	Mumbai	6500