**Ex. No: 7 Date: 08 – 09 - 2022**

**Experiment 7**

Implicit Cursors

**Consider the following schema of a database:**

**employees** (employeeId, employeeName, employeeCity)

**Tables:**

**employees**

|  |  |  |
| --- | --- | --- |
| employeeId | employeeName | employeeCity |
| 1 | XXX | Chennai |
| 2 | XYZ | Mumbai |
| 3 | YYY | Calcutta |

**Write SQL queries to**

* Delete record having employeeId = 2, and if successful, insert values (2, 'ZZZ', 'Delhi').
* Do previous query but instead of inserting values, update employeeCity = Chandigarh where employeeId = 1.
* Update employeeCity = Gurgaon where employeeId = 2, and if successful, delete record having employeeId = 1.
* Do first query, but instead of inserting values, just show number of employees deleted.

**Queries:**

* Delete record having employeeId = 2, and if successful, insert values (2, 'ZZZ', 'Delhi').

CREATE TABLE tempory\_employee AS SELECT \* FROM employees;

DECLARE

employeeNo NUMBER(4) := 2;

BEGIN

DELETE FROM tempory\_employee WHERE employeeId = employeeNo ;

IF SQL%FOUND THEN -- delete succeeded

INSERT INTO tempory\_employee (employeeId,employeeName,employeeCity) VALUES (2, 'ZZZ', 'Delhi');

END IF;

END;

/

* Do previous question but instead of inserting values, update employeeCity = Chandigarh where employeeId = 1.

CREATE TABLE tempory\_employee1 AS SELECT \* FROM employees;

DECLARE

employeeNo NUMBER(4) := 2;

BEGIN

DELETE FROM tempory\_employee WHERE employeeId = employeeNo ;

IF SQL%FOUND THEN -- delete succeeded

UPDATE employees SET employeeCity = 'Chandigarh' WHERE employeeId = 1;

END IF;

END;

/

* Update employeeCity = Gurgaon where employeeId = 2, and if successful, delete record having employeeId = 1.

CREATE TABLE tempory\_employee2 AS SELECT \* FROM employees;

DECLARE

employeeNo NUMBER(4) := 2;

BEGIN

UPDATE tempory\_employee2 SET employeeCity = 'Gurgaon' WHERE employeeId = employeeNo;

IF SQL%FOUND THEN -- update succeeded

DELETE FROM tempory\_employee2 WHERE employeeId = 1 ; -- Then delete a specific row

END IF;

END;

/

* Do Query 1, but instead of inserting values, just show number of employees deleted.

CREATE TABLE tempory\_employee3 AS SELECT \* FROM employees;

DECLARE employeeNo NUMBER(4) := 2;

BEGIN

DELETE FROM tempory\_employee3 WHERE employeeId = employeeNo ;

DBMS\_OUTPUT.PUT\_LINE('Number of employees deleted: ' || TO\_CHAR(SQL%ROWCOUNT));

END;

**Code:**

SQL> create table employees (employeeid number(10) not null, employeename varchar2(50) not null, employeecity varchar2(50), primary key (employeeid));

Table created.

SQL> desc employees;

Name Null? Type

----------------------------------------- -------- -----------------------

EMPLOYEEID NOT NULL NUMBER(10)

EMPLOYEENAME NOT NULL VARCHAR2(50)

EMPLOYEECITY VARCHAR2(50)

SQL> insert into employees values(1,'XXX','Chennai');

1 row created.

SQL> insert into employees values(2,'XYZ','Mumbai');

1 row created.

SQL> insert into employees values(3,'YYY','Calcutta');

1 row created.

SQL> select \* from employees;

EMPLOYEEID EMPLOYEENAME

---------- --------------------------------------------------

EMPLOYEECITY

--------------------------------------------------

1 XXX

Chennai

2 XYZ

Mumbai

3 YYY

Calcutta

SQL> set linesize 1500;

SQL> select \* from employees;

EMPLOYEEID EMPLOYEENAME EMPLOYEECITY

---------- -------------------------------------------------- -------------

1 XXX Chennai

2 XYZ Mumbai

3 YYY Calcutta

SQL> set serveroutput on;

SQL> @ C:\Users\2162014\DBMS\temp\_emp.sql;

Table created.

PL/SQL procedure successfully completed.

SQL> select \* from tempory\_employee;

EMPLOYEEID EMPLOYEENAME EMPLOYEECITY

---------- -------------------------------------------------- -------------

1 XXX Chennai

3 YYY Calcutta

2 ZZZ Delhi

SQL> @ C:\Users\2162014\DBMS\temp\_emp1.sql;

Table created.

PL/SQL procedure successfully completed.

SQL> select \* from tempory\_employee1;

EMPLOYEEID EMPLOYEENAME EMPLOYEECITY

---------- -------------------------------------------------- -------------

1 XXX Chennai

2 XYZ Mumbai

3 YYY Calcutta

SQL> @ C:\Users\2162014\DBMS\temp\_emp2.sql;

Table created.

PL/SQL procedure successfully completed.

SQL> select \* from tempory\_employee2;

EMPLOYEEID EMPLOYEENAME EMPLOYEECITY

---------- -------------------------------------------------- -------------

2 XYZ Gurgaon

3 YYY Calcutta

SQL> @ C:\Users\2162014\DBMS\temp\_emp3.sql;

Table created.

PL/SQL procedure successfully completed.

SQL> select \* from tempory\_employee3;

EMPLOYEEID EMPLOYEENAME EMPLOYEECITY

---------- -------------------------------------------------- ------------- 1 XXX Chandigarh 3 YYY Calcutta

Ex. No: 8 Date: 27 – 10 - 2022

**Experiment 8**

Explicit Cursors

**Consider the following schema of a database:**

**customers** (ID, NAME, AGE, ADDRESS, SALARY)

**Table:**

**customers**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | NAME | AGE | ADDRESS | SALARY |
| 1 | Ramesh | 32 | Ahmedabad | 2500.00 |
| 2 | Khilan | 25 | Delhi | 2000.00 |
| 3 | Kaushik | 23 | Kota | 2500.00 |
| 4 | Chaitali | 25 | Mumbai | 7000.00 |
| 5 | Hardik | 27 | Bhopal | 9000.00 |
| 6 | Komal | 22 | MP | 5000.00 |

**Write a SQL query to display customer’s ID, Name and Address using explicit cursors**

Query

DECLARE

c\_id customers.id%type;

c\_name customers.name%type;

c\_addr customers.address%type;

CURSOR c\_customers is

SELECT id, name, address FROM customers;

BEGIN

OPEN c\_customers;

LOOP

FETCH c\_customers into c\_id, c\_name, c\_addr;

EXIT WHEN c\_customers%notfound;

dbms\_output.put\_line(c\_id || ' ' || c\_name || ' ' || c\_addr);

END LOOP;

CLOSE c\_customers;

END;

/

**Code:**

SQL> create table Customers (ID number(1), Name varchar2(20), Age number(2), Address varchar2(20), Salary float, primary key(ID));

Table created.

SQL> insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary);

Enter value for id: 1

Enter value for name: Ramesh

Enter value for age: 32

Enter value for address: Ahmedabad

Enter value for salary: 2000

old 1: insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary)

new 1: insert into Customers values(1,'Ramesh', 32,'Ahmedabad', 2000)

1 row created.

SQL> insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary);

Enter value for id: 2

Enter value for name: Khilan

Enter value for age: 25

Enter value for address: Delhi

Enter value for salary: 1500

old 1: insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary)

new 1: insert into Customers values(2,'Khilan', 25,'Delhi', 1500)

1 row created.

SQL> insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary);

Enter value for id: 3

Enter value for name: Kaushik

Enter value for age: 23

Enter value for address: Kota

Enter value for salary: 2000

old 1: insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary)

new 1: insert into Customers values(3,'Kaushik', 23,'Kota', 2000)

1 row created.

SQL> insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary);

Enter value for id: 4

Enter value for name: Chaitali

Enter value for age: 25

Enter value for address: Mumbai

Enter value for salary:

old 1: insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary)

new 1: insert into Customers values(4,'Chaitali', 25,'Mumbai', )

insert into Customers values(4,'Chaitali', 25,'Mumbai', )

\*

ERROR at line 1:

ORA-00936: missing expression

SQL> insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary);

Enter value for id: 4

Enter value for name: Chaitali

Enter value for age: 25

Enter value for address: Mumbai

Enter value for salary: 6500

old 1: insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary)

new 1: insert into Customers values(4,'Chaitali', 25,'Mumbai', 6500)

1 row created.

SQL> insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary);

Enter value for id: 5

Enter value for name: Hardik

Enter value for age: 27

Enter value for address: Bhopal

Enter value for salary: 8500

old 1: insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary)

new 1: insert into Customers values(5,'Hardik', 27,'Bhopal', 8500)

1 row created.

SQL> insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary);

Enter value for id: 6

Enter value for name: Komal

Enter value for age: 22

Enter value for address: MP

Enter value for salary: 4500

old 1: insert into Customers values(&ID,'&Name', &Age,'&Address', &Salary)

new 1: insert into Customers values(6,'Komal', 22,'MP', 4500)

1 row created.

SQL> desc customers

Name Null? Type

----------------------------------------- -------- ----------------------------

ID NOT NULL NUMBER(1)

NAME VARCHAR2(20)

AGE NUMBER(2)

ADDRESS VARCHAR2(20)

SALARY FLOAT(126)

SQL> select \* from customers;

ID NAME AGE ADDRESS SALARY

---------- -------------------- ---------- -------------------- ----------

1 Ramesh 32 Ahmedabad 3000

2 Khilan 25 Delhi 2500

3 Kaushik 23 Kota 3000

4 Chaitali 25 Mumbai 7500

5 Hardik 27 Bhopal 9500

6 Komal 22 MP 5500

6 rows selected.

SQL> set serveroutput on

SQL> DECLARE

2 c\_id customers.id%type;

3 c\_name customers.name%type;

4 c\_addr customers.address%type;

5 CURSOR c\_customers is

6 SELECT id, name, address FROM customers;

7 BEGIN

8 OPEN c\_customers;

9 LOOP

10 FETCH c\_customers into c\_id, c\_name, c\_addr;

11 EXIT WHEN c\_customers%notfound;

12 dbms\_output.put\_line(c\_id || ' ' || c\_name || ' ' || c\_addr);

13 END LOOP;

14 CLOSE c\_customers;

15 END;

16 /

1 Ramesh Ahmedabad

2 Khilan Delhi

3 Kaushik Kota

4 Chaitali Mumbai

5 Hardik Bhopal

6 Komal MP

PL/SQL procedure successfully completed.