PRACTICAL – 07

NAME : Aryan Kashikar
Roll n0: D2 32

Practical 7 : To implement Cursor in Oracle 11g
EMP
EMPNO ENAME JOB MGR HIREDATE SAL COMM DEPTNO

```
SQL> select * from empl;

     EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
---------- ---------- --------- ---------- --------- ---------- ---------- ----------
      7369 SMITH      CLERK           7902 17-DEC-80        800                    20
      7499 ALLEN      SALESMAN        7698 20-FEB-81       1600        300         30
      7521 WARD       SALESMAN        7698 22-FEB-81       1250        500         30
      7566 JONES      MANAGER         7839 02-APR-81       2975                    20
      7654 MARTIN     SALESMAN        7698 28-SEP-81       1250       1400         30
      7698 BLAKE      MANAGER         7839 01-MAY-81       2850                    30
      7782 CLARK      MANAGER         7839 09-JUN-81       2450                    10
      7788 SCOTT      ANALYST         7566 19-APR-87       3000                    20
      7839 KING       PRESIDENT            17-NOV-81       5000                    10
      7844 TURNER     SALESMAN        7698 08-SEP-81       1500          0         30
      7876 ADAMS      CLERK           7788 23-MAY-87       1100                    20
      7900 JAMES      CLERK           7698 03-DEC-81        950                    30
      7902 FORD       ANALYST         7566 03-DEC-81       3000                    20
      7934 MILLER     CLERK           7782 23-JAN-82       1300                    10

14 rows selected.
```

DEPT
DEPTNO DNAME LOC

```
SQL> select * from deptm;

    DEPTNO DNAME          LOC
---------- -------------- -------------
        10 ACCOUNTING     NEW YORK
        20 RESEARCH       DALLAS
        30 SALES          CHICAGO
        40 OPERATIONS     BOSTON
```

1. Increment salary of employee by 40% where dept name starts with „A" and joining date is in
year 2000. Find the number of rows affected by the use of SQL%ROWCOUNT attributes of
an implicit cursor. (implicit cursor). Also print message “Records not found" if no such data exist.

```
DECLARE
  v_dept_name dept.dname%TYPE := 'A%';
  v_year NUMBER := 2000;
BEGIN
  UPDATE empl
  SET sal = sal * 1.4
  WHERE deptno IN (
    SELECT deptno
    FROM deptm
    WHERE dname LIKE v_dept_name
  )
  AND EXTRACT(YEAR FROM hiredate) = v_year;

  IF SQL%ROWCOUNT = 0 THEN
    DBMS_OUTPUT.PUT_LINE('Records not found');
  ELSE
    DBMS_OUTPUT.PUT_LINE('Number of rows affected: ' || SQL%ROWCOUNT);
  END IF;
END;
/
```

```
SQL> @E:\Coding\DBMSPract7\01.sql
Records not found

PL/SQL procedure successfully completed.
```

2. Create table emp1(empno,name,sal) emp2(empno,name,sal) . Initially emp1, emp2 are empty.Define explicit cursor on table emp. (select all columns using %rowtype). Insert records of deptno(10,20) in table emp1. And Insert other records in table emp2.

```
DECLARE
  CURSOR c_emp IS
    SELECT *
    FROM empl;
  v_emp c_emp%ROWTYPE;
BEGIN
  OPEN c_emp;
  LOOP
    FETCH c_emp INTO v_emp;
    EXIT WHEN c_emp%NOTFOUND;

    IF v_emp.deptno IN (10, 20) THEN
      INSERT INTO emp1(empno,ename, sal)
      VALUES (v_emp.empno, v_emp.ename, v_emp.sal);
    ELSE
```

```
      INSERT INTO emp2(empno,ename, sal)
      VALUES (v_emp.empno, v_emp.ename, v_emp.sal);
    END IF;
  END LOOP;
  CLOSE c_emp;
END;
/
```

```
SQL> @E:\Coding\DBMSPract7\02.sql

PL/SQL procedure successfully completed.

SQL> select * from emp1;

     EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
---------- ---------- --------- ---------- --------- ---------- ---------- ----------
      7369 SMITH                                          800
      7566 JONES                                         2975
      7782 CLARK                                         2450
      7788 SCOTT                                         3000
      7839 KING                                          5000
      7876 ADAMS                                         1100
      7902 FORD                                          3000
      7934 MILLER                                        1300

8 rows selected.

SQL> select * from emp2;

     EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
---------- ---------- --------- ---------- --------- ---------- ---------- ----------
      7499 ALLEN                                         1600
      7521 WARD                                          1250
      7654 MARTIN                                        1250
      7698 BLAKE                                         2850
      7844 TURNER                                        1500
      7900 JAMES                                          950

6 rows selected.
```

3. Write a program in PL/SQL to FETCH multiple records and more than one columns from different tables.

Define cursor to fetch data ename,sal,dname using table emp and dept. Display data where sal>3000.

```
DECLARE
  CURSOR c_emp_dept IS
    SELECT e.ename, e.sal, d.dname
    FROM empl e
    JOIN deptm d ON e.deptno = d.deptno
    WHERE e.sal > 3000;
  v_ename empl.ename%TYPE;
  v_sal empl.sal%TYPE;
  v_dname deptm.dname%TYPE;
BEGIN
  OPEN c_emp_dept;
  LOOP
    FETCH c_emp_dept INTO v_ename, v_sal, v_dname;
```

```
    EXIT WHEN c_emp_dept%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE(v_ename || ' ' || v_sal || ' ' || v_dname);
  END LOOP;
  CLOSE c_emp_dept;
END;
/
```

```
SQL> @E:\Coding\DBMSPract7\03.sql
KING 5000 ACCOUNTING

PL/SQL procedure successfully completed.
```

4. Write a program in PL/SQL to create a cursor displays the name and salary of each employee
in the EMPLOYEES table whose department number is equal to parameter 1, salary is less
than a parameter2 value passed in cursor. (use parameterized cursor).

```
DECLARE
  CURSOR c_employee(p_deptno NUMBER, p_salary NUMBER) IS
    SELECT ename, sal
    FROM empl
    WHERE deptno = p_deptno
    AND sal < p_salary;
  v_ename empl.ename%TYPE;
  v_sal empl.sal%TYPE;
BEGIN
  OPEN c_employee(10, 5000);
  LOOP
    FETCH c_employee INTO v_ename, v_sal;
    EXIT WHEN c_employee%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE(v_ename || ' ' || v_sal);
  END LOOP;
  CLOSE c_employee;
END;
/
```

```
PL/SQL procedure successfully completed.

SQL> @E:\Coding\DBMSPract7\04.sql
CLARK 2450
MILLER 1300

PL/SQL procedure successfully completed.
```

5. Define cursor on emp to store data of employs those worked for more than 1 year in the organization. Increament salary of these employees by 30%.

```
DECLARE
  CURSOR c_emp IS
    SELECT *
    FROM empl
    WHERE hiredate < SYSDATE - 365;
BEGIN
  FOR emp_rec IN c_emp LOOP
    UPDATE empl
    SET sal = sal * 1.3
    WHERE empno = emp_rec.empno;
  END LOOP;
  COMMIT;
END;
/
```

```
SQL> select * from empl;

     EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO
---------- ---------- --------- ---------- --------- ---------- ---------- ----------
      7369 SMITH      CLERK           7902 17-DEC-80       1040                    20
      7499 ALLEN      SALESMAN        7698 20-FEB-81       2080        300         30
      7521 WARD       SALESMAN        7698 22-FEB-81       1625        500         30
      7566 JONES      MANAGER         7839 02-APR-81     3867.5                    20
      7654 MARTIN     SALESMAN        7698 28-SEP-81       1625       1400         30
      7698 BLAKE      MANAGER         7839 01-MAY-81       3705                    30
      7782 CLARK      MANAGER         7839 09-JUN-81       3185                    10
      7788 SCOTT      ANALYST         7566 19-APR-87       3900                    20
      7839 KING       PRESIDENT            17-NOV-81       6500                    10
      7844 TURNER     SALESMAN        7698 08-SEP-81       1950          0         30
      7876 ADAMS      CLERK           7788 23-MAY-87       1430                    20
      7900 JAMES      CLERK           7698 03-DEC-81       1235                    30
      7902 FORD       ANALYST         7566 03-DEC-81       3900                    20
      7934 MILLER     CLERK           7782 23-JAN-82       1690                    10

14 rows selected.
```

6. Create table account. Define cursor on account to select data of acct no 11,12,13. Fetch cursor data to withdraw amount of Rs 2000. If Minimum balance after deduction >=1000, then only deduct from table account otherwise Insert data of failure transaction “insufficient fund” in table failtransaction. [refer to pdf sent]

```
CREATE TABLE account (
  acct_no NUMBER,
  balance NUMBER
);

CREATE TABLE failtransaction (
```

```
  message VARCHAR2(100)
);

INSERT INTO account(acct_no, balance)
VALUES (11, 5000);

INSERT INTO account(acct_no, balance)
VALUES (12, 3000);

INSERT INTO account(acct_no, balance)
VALUES (13, 8000);

DECLARE
  CURSOR c_account IS
    SELECT *
    FROM account
    WHERE acct_no IN (11, 12, 13);
  v_acct_no account.acct_no%TYPE;
  v_balance account.balance%TYPE;
BEGIN
  FOR account_rec IN c_account LOOP
    v_acct_no := account_rec.acct_no;
    v_balance := account_rec.balance;

    IF v_balance - 2000 >= 1000 THEN
      UPDATE account
      SET balance = balance - 2000
      WHERE acct_no = v_acct_no;
    ELSE
      INSERT INTO failtransaction(message)
      VALUES ('insufficient funds');
    END IF;
  END LOOP;
  COMMIT;
END;
/
```

```
SQL> @E:\Coding\DBMSPract7\06.sql

Table created.


Table created.


1 row created.


1 row created.


1 row created.


PL/SQL procedure successfully completed.

SQL> select * from account;

   ACCT_NO    BALANCE
---------- ----------
        11       3000
        12       1000
        13       6000
SQL> select * from failtransaction;

no rows selected

SQL> @E:\Coding\DBMSPract7\06.sql

PL/SQL procedure successfully completed.

SQL> select * from failtransaction;

MESSAGE
--------------------------------------------------------------------------------
insufficient funds
```

7. Define cursor to select deptno,dname and employee count working in that department. (use
join emp and dept). Use for loop and Display all information.

```
DECLARE
  CURSOR c_dept_employee IS
    SELECT d.deptno, d.dname, COUNT(*) AS emp_count
    FROM deptm d
    JOIN empl e ON d.deptno = e.deptno
    GROUP BY d.deptno, d.dname;
BEGIN
  FOR dept_employee_rec IN c_dept_employee LOOP
```

```
    DBMS_OUTPUT.PUT_LINE('Deptno: ' || dept_employee_rec.deptno || ', Dname: ' ||
dept_employee_rec.dname || ', Employee Count: ' || dept_employee_rec.emp_count);
  END LOOP;
END;
/
```

```
SQL> @E:\Coding\DBMSPract7\07.sql
Deptno: 10, Dname: ACCOUNTING, Employee Count: 3
Deptno: 20, Dname: RESEARCH, Employee Count: 5
Deptno: 30, Dname: SALES, Employee Count: 6

PL/SQL procedure successfully completed.
```

8. Modify 7 to Define cursor to select deptno,dname and employee count working in that department having employee count >=3. (use join emp and dept). Use for loop and Display all information.

```
DECLARE
  CURSOR c_dept_employee IS
    SELECT d.deptno, d.dname, COUNT(*) AS emp_count
    FROM deptm d
    JOIN empl e ON d.deptno = e.deptno
    GROUP BY d.deptno, d.dname
    HAVING COUNT(*) >= 3;
BEGIN
  FOR dept_employee_rec IN c_dept_employee LOOP
    DBMS_OUTPUT.PUT_LINE('Deptno: ' || dept_employee_rec.deptno || ', Dname: ' ||
dept_employee_rec.dname || ', Employee Count: ' || dept_employee_rec.emp_count);
  END LOOP;
END;
/
```

```
SQL> @E:\Coding\DBMSPract7\08.sql
Deptno: 10, Dname: ACCOUNTING, Employee Count: 3
Deptno: 20, Dname: RESEARCH, Employee Count: 5
Deptno: 30, Dname: SALES, Employee Count: 6

PL/SQL procedure successfully completed.
```