

NAME:- SHUBHAM SAXENA

REG No:- RA1811003010691

SEC:- D2

DATE:- 27/4/2021

DDL WORKSHEET-XIII

I. Write a trigger to ensure that no employee of age less than 25 can be inserted in the database.

```
F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> CREATE TRIGGER CHECK_AGE BEFORE INSERT ON EMP
2 FOR EACH ROW
3 BEGIN
4 IF NEW.AGE < 25 THEN
5 SIGNAL SQLSTATE '45000'
6 SET MESSAGE_TEXT = 'ERROR:
7 AGE MUST BE AT LEAST 25 YEARS!';
8 END IF;
9 END;
10 /
Warning: Trigger created with compilation errors.
```

II. Create a trigger which will work before deletion in employee table and create a duplicate copy of the record in another table employee_old.

```
F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> create table employee_backup(empno number(5),
2  ename varchar(20), job varchar(40),
3  hiredate date, salary number(8),
4  primary key(empno));
Table created.
SQL> create trigger backup before delete on emp
2 for each row
3 begin
4 insert into employee_backup
5 values (emp.empno, emp.ename, emp.job, emp.hiredate, emp.salary);
6 end;
7 /
Warning: Trigger created with compilation errors.
```

III. Write a trigger to count number of new tuples inserted using each insert statement.

```
F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> CREATE OR REPLACE TRIGGER trg
2 AFTER INSERT
3 ON EMP
4 FOR EACH ROW
5
6 DECLARE
7 val number;
8 BEGIN
9 SELECT COUNT(*)
10 INTO val
11 FROM EMP;
12
13 DBMS_OUTPUT.PUT_LINE('TABLE COUNT AFTER INSERT = '||val);
14
15 END;
16 /
Trigger created.
```

IV. Write a program in PL/SQL to find the number of rows effected by the use of SQL%ROWCOUNT attributes of an implicit cursor.

```
F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> set serveroutput on;
SQL> CREATE TABLE emp_temp AS
2 SELECT empno, ename, job FROM emp;
Table created.
SQL> BEGIN
2 UPDATE emp_temp
3 SET job = 'not available'
4 WHERE ename LIKE 'B%';
5
6 dbms_output.put_line('Number of record updated: '
7 ||To_char(SQL%rowcount));
8 END;
9 /
Number of record updated: 0
PL/SQL procedure successfully completed.
```

```

F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> CREATE TABLE emp_temp AS
2   SELECT empno, ename, job FROM emp;
CREATE TABLE emp_temp AS
*
ERROR at line 1:
ORA-00955: name is already used by an existing object

SQL> BEGIN
2   UPDATE emp_temp
3   SET    job = 'not available'
4   WHERE  ename LIKE 'D%';
5
6   dbms_output.Put_line('Number of record updated: '
7   ||to_char(SQL%rowcount));
8 END;
9 /
Number of record updated: 1
PL/SQL procedure successfully completed.

```

V. Write a program in PL/SQL to show the uses of implicit cursor without using any attribute.

```

F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> DECLARE
2   emp_ename VARCHAR2(35);
3   zemp_id NUMBER:=&empno;
4 BEGIN
5   SELECT ename INTO emp_ename FROM emp WHERE empno = zemp_id;
6   dbms_output.Put_line ('Employee name: '
7   || emp_ename);
8 EXCEPTION
9   WHEN no_data_found THEN
10    dbms_output.Put_line ('There is no employee with the ID '||to_char(zemp_id
11);
12 END;
13 /
Enter value for empno: 18
old 3: zemp_id NUMBER:=&empno;
new 3: zemp_id NUMBER:=18;
Employee name: Eshu

```

VI. Write a program in PL/SQL to retrieve the records from the employees table and display them using cursors.

```

F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> DECLARE
2   z_empid emp.empno%TYPE;
3   z_empname emp.ename%TYPE;
4   z_salary emp.salary%TYPE;
5   CURSOR employee_cursor IS -- declaring a cursor
6   SELECT empno,
7   ename,
8   salary
9   FROM emp;
10
11 BEGIN
12   OPEN employee_cursor; -- opening the cursor
13   LOOP
14     FETCH employee_cursor -- fetching records from the cursor
15     INTO z_empid,
16     z_empname,
17     z_salary;
18   EXIT
19   WHEN employee_cursor%NOTFOUND;
20   IF (z_salary > 8000) THEN
21     dbms_output.Put_line(z_empid
22     || ' '
23     || z_empname
24     || ' '
25     || z_salary);
26   ELSE
27     dbms_output.Put_line(z_empname
28     || ' salary is less then 8000');
29   END IF;
30   END LOOP;
31   CLOSE employee_cursor; --closing the cursor
32 END;
33 /
4 Dev 10000
6 Shubham 10000
Ajay salary is less then 8000
Ankita salary is less then 8000
Prachi salary is less then 8000
Utkarsh salary is less then 8000
Shiv salary is less then 8000
Eshu salary is less then 8000

```

VII. Create a PL/SQL block to increase salary of employees in the department 20 using explicit cursor.

```
F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> DECLARE
2  CURSOR emp2 IS
3  SELECT empno,
4  salary
5  FROM emp
6  WHERE deptno = 20
7  FOR UPDATE;
8  incr_sal NUMBER;
9  BEGIN
10  FOR emp IN emp2 LOOP
11  IF emp.salary < 15000 THEN
12  incr_sal := .15;
13  ELSE
14  incr_sal := .10;
15  END IF;
16  UPDATE emp
17  SET salary = salary + salary * incr_sal
18  WHERE CURRENT OF emp2;
19  END LOOP;
20  END;
21  /

PL/SQL procedure successfully completed.
```

VIII. Write a program in PL/SQL to print a list of managers and the name of the departments (explicit).

```
F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> DECLARE
2  CURSOR cur_mgr IS
3  SELECT ENAME,
4  DEP
5  FROM emp e
6  INNER JOIN DEPT d ON d.ID = e.DEPTNO;
7
8  v_mgr cur_mgr%ROWTYPE;
9  BEGIN
10  OPEN cur_mgr;
11  LOOP
12  FETCH cur_mgr INTO v_mgr;
13  EXIT WHEN cur_mgr%NOTFOUND;
14  DBMS_OUTPUT.PUT_LINE(v_mgr.DEP || ' :: ' ||
15  v_mgr.ENAME);
16  END LOOP;
17  CLOSE cur_mgr;
18  END;
19  /
Analysis :: Shubham
Analysis :: Ajay
Analysis :: Ankita
Testing :: Prachi

PL/SQL procedure successfully completed.
```

XI. Write a program in PL/SQL to create a cursor displays the name and salary of each employee in the EMPLOYEES table whose salary is less than that specified by a passed-in parameter value.

```
F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> DECLARE
2  ename emp.ename%type;
3  salary emp.salary%type;
4  CURSOR emp1 is
5  SELECT ename,salary FROM emp where salary < 25000;
6  BEGIN
7  OPEN emp1;
8  LOOP
9  FETCH emp1 into ename,salary;
10  EXIT WHEN emp1%notfound;
11  dbms_output.put_line(ename || ' ' ||salary);
12  END LOOP;
13  CLOSE emp1;
14  END;
15  /
Dev 11500
Shubham 10000
Ajay 5000
Ankita 4000
Prachi 8000
Utkarsh 8050
Shiv 6325
Eshu 8050

PL/SQL procedure successfully completed.
```

X. Write a PL/SQL block to find the name and salary of first five highest paid salaries using explicit cursor.

```
F:\Downloads\ORACLE CLIENT 11.2\ORACLE CLIENT 11.2\instantclient_11_2\sqlplus.exe
SQL> DECLARE
2   empno emp.empno%type;
3   ename emp.ename%type;
4   salary emp.salary%type;
5   CURSOR e_emp is
6     SELECT empno,ename,salary FROM emp order by salary desc;
7   I NUMBER;
8 BEGIN
9   OPEN e_emp;
10  FOR I IN 1..5
11  LOOP
12    FETCH e_emp into empno,ename,salary;
13    EXIT WHEN e_emp%notfound;
14    dbms_output.put_line(empno|| ' ' || ename || ' ' || salary);
15  END LOOP;
16  CLOSE e_emp;
17 END;
18 /
4 Dev 11500
6 Shubham 10000
18 Eshu 8050
12 Utkarsh 8050
1 Prachi 8000
PL/SQL procedure successfully completed.
```