Assignments(PLSQL)

1. Write a procedure for the following.

a. To accept employee number , delete the record of the given

employee.

```
  1  create or replace procedure insert_into(eno number)
  2  as
  3  begin
  4  delete from emp where empno=eno;
  5* end;
SQL> /

Procedure created.
```

BEFORE

```
     EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO     NETSAL
---------- ---------- --------- ---------- --------- ---------- ---------- ---------- ----------
      7369 SMITH      CLERK           7902 17-DEC-80        800                    20       1000
      7499 ALLEN      SALESMAN        7698 20-FEB-81       1600        300         30       1000
      7521 WARD       SALESMAN        7698 22-FEB-81       1250        500         30       1000
      7566 JONES      MANAGER         7839 02-APR-81       2975                    20       1000
      7654 MARTIN     SALESMAN        7698 28-SEP-81       1250       1400         30       1000
      7698 BLAKE      MANAGER         7839 01-MAY-81       2850                    30       1000
      7782 CLARK      MANAGER         7839 09-JUN-81       2450                    10       1000
      7788 SCOTT      ANALYST         7566 09-DEC-82       3000                    20       1000
      7839 KING       PRESIDENT            17-NOV-81       5000                    10       1000
      7844 TURNER     SALESMAN        7698 08-SEP-81       1500          0         30       1000
      7876 ADAMS      CLERK           7788 12-JAN-83       1100                    20       1000
      7900 JAMES      CLERK           7698 03-DEC-81        950                    30       1000
      7902 FORD       ANALYST         7566 03-DEC-81       3000                    20       1000
      7934 MILLER     CLERK           7782 23-JAN-82       1300                    10       1000

14 rows selected.
```

AFTER

```
SQL> execute insert_into(7369);

PL/SQL procedure successfully completed.

SQL> select * from emp;

     EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM     DEPTNO     NETSAL
---------- ---------- --------- ---------- --------- ---------- ---------- ---------- ----------
      7499 ALLEN      SALESMAN        7698 20-FEB-81       1600        300         30       1000
      7521 WARD       SALESMAN        7698 22-FEB-81       1250        500         30       1000
      7566 JONES      MANAGER         7839 02-APR-81       2975                    20       1000
      7654 MARTIN     SALESMAN        7698 28-SEP-81       1250       1400         30       1000
      7698 BLAKE      MANAGER         7839 01-MAY-81       2850                    30       1000
      7782 CLARK      MANAGER         7839 09-JUN-81       2450                    10       1000
      7788 SCOTT      ANALYST         7566 09-DEC-82       3000                    20       1000
      7839 KING       PRESIDENT            17-NOV-81       5000                    10       1000
      7844 TURNER     SALESMAN        7698 08-SEP-81       1500          0         30       1000
      7876 ADAMS      CLERK           7788 12-JAN-83       1100                    20       1000
      7900 JAMES      CLERK           7698 03-DEC-81        950                    30       1000
      7902 FORD       ANALYST         7566 03-DEC-81       3000                    20       1000
      7934 MILLER     CLERK           7782 23-JAN-82       1300                    10       1000

13 rows selected.
```

b. To accept grade,losal and hisal and insert a record into

salgrad table

```
  1  create or replace procedure insert_into(g in number, l in number, h in number)
  2  as
  3  begin
  4  insert into salgrade
  5  values(g,l,h);
  6* end;
SQL> /

Procedure created.

SQL> select * from salgrade;

    GRADE      LOSAL      HISAL
---------- ---------- ----------
         1        700       1200
         2       1201       1400
         3       1401       2000
         4       2001       3000
         5       3001       9999

SQL> execute insert_into(6,10000,15000);

PL/SQL procedure successfully completed.

SQL> select * from salgrade;

    GRADE      LOSAL      HISAL
---------- ---------- ----------
         1        700       1200
         2       1201       1400
         3       1401       2000
         4       2001       3000
         5       3001       9999
         6      10000      15000

6 rows selected.
```

2. Write functions to perform the following.

a. Calculate experience of the employee

```
  1  create or replace function calc_exp(h date)
  2  return number
  3  as
  4  begin
  5  return round(months_between(sysdate,h)/12);
  6* end;
SQL> /

Function created.
```

```
  1  select hiredate,calc_exp(hiredate)
  2* from emp
SQL> /

HIREDATE  CALC_EXP(HIREDATE)
--------- ------------------
20-FEB-81                 43
22-FEB-81                 43
02-APR-81                 43
28-SEP-81                 42
01-MAY-81                 42
09-JUN-81                 42
09-DEC-82                 41
17-NOV-81                 42
08-SEP-81                 42
12-JAN-83                 41
03-DEC-81                 42
03-DEC-81                 42
23-JAN-82                 42

13 rows selected.

SQL>
```

b. Ti calculate net sal by using formula.

Netsal=sal+da+hra-pf+comm

Da-→ 10% sal hra→ 15% sal pf --→ 8 % of sal

```
  1  create or replace function net_sal(salary number,co number)
  2  return number
  3  as
  4  begin
  5     return salary*1.17+nvl(co,0);
  6* end;
SQL> /

Function created.

SQL>
```

```
  1  select empno,ename,sal,nvl(comm,0) comm,net_sal(sal,comm) net_salary
  2* from emp
SQL> /

     EMPNO ENAME             SAL       COMM NET_SALARY
---------- ---------- ---------- ---------- ----------
      7521 WARD             1250        500     1962.5
      7566 JONES            2975          0    3480.75
      7654 MARTIN           1250       1400     2862.5
      7698 BLAKE            2850          0     3334.5
      7782 CLARK            2450          0     2866.5
      7788 SCOTT            3000          0       3510
      7839 KING             5000          0       5850
      7844 TURNER           1500          0       1755
      7876 ADAMS            1100          0       1287
      7900 JAMES            1234          0    1443.78
      7902 FORD             3000          0       3510
      7934 MILLER           1300          0       1521

12 rows selected.

SQL>
```

Loops example

1. Print the following patterns using loop :

a.

*

**

***

****

```
  1  declare
  2  num number:=&n;
  3  i number;
  4  j number;
  5  begin
  6  for i in 1..num
  7  loop
  8     for j in 1..i
  9     loop
 10             DBMS_OUTPUT.PUT('*');
 11     end loop;
 12  DBMS_OUTPUT.NEW_LINE;
 13  end loop;
 14* end;
SQL> /
Enter value for n: 4
old   2: num number:=&n;
new   2: num number:=4;
*
**
***
****
```

b.

```
  *

 ***

*****

 ***

  *
```

```
 1  create or replace procedure diamond
 2  as
 3  i number;
 4  j number;
 5  k number;
 6  l number;
 7  begin
 8  for i in 1..3
 9  loop
10     k:=2*i-1;
11     l:=(5-k)/2;
12     for j in reverse 1..l
13     loop
14             DBMS_OUTPUT.PUT(' ');
15     end loop;
16     for j in 1..k
17     loop
18             DBMS_OUTPUT.PUT('*');
19     end loop;
20     DBMS_OUTPUT.NEW_LINE;
21  end loop;
22  for i in reverse 1..2
23  loop
24     k:=2*i-1;
25     l:=(5-k)/2;
26     for j in reverse 1..l
27     loop
28             DBMS_OUTPUT.PUT(' ');
29     end loop;
30     for j in 1..k
31     loop
32             DBMS_OUTPUT.PUT('*');
33     end loop;
34     DBMS_OUTPUT.NEW_LINE;
35  end loop;
36* end;
QL> /
```

```
Procedure created.

SQL> execute diamond();
  *
 ***
*****
 ***
  *

PL/SQL procedure successfully completed.
```

ANOTHER METHOD

```
 1  create or replace procedure diamond
 2  as
 3  i number;
 4  j number;
 5  k number;
 6  m number;
 7  star varchar2(20);
 8  str varchar2(20);
 9  begin
10     for i in 1..3
11     loop
12     k:=2*i;
13     m:=4-i;
14     str:=lpad(' ',k,'*');
15     star:=concat(lpad(' ',m,' '),str);
16     DBMS_OUTPUT.PUT_LINE(star);
17     end loop;
18     for i in reverse 1..2
19     loop
20     k:=2*i;
21     m:=4-i;
22     str:=lpad(' ',k,'*');
23     star:=concat(lpad(' ',m,' '),str);
24     DBMS_OUTPUT.PUT_LINE(star);
25     end loop;
26* end;
SQL> /

Procedure created.

SQL> execute diamond();
   *
  ***
 *****
  ***
   *
```

c.

1010101

10101

101

1

```
 1  create or replace procedure one
 2  as
 3  i number;
 4  j number;
 5  begin
 6  for i in reverse 1..3
 7  loop
 8     for j in 1..i
 9     loop
10             DBMS_OUTPUT.PUT('10');
11     end loop;
12     DBMS_OUTPUT.PUT('1');
13     DBMS_OUTPUT.NEW_LINE;
14  end loop;
15  DBMS_OUTPUT.PUT_LINE('1');
16* end;
SQL> /

Procedure created.

SQL> execute one()
1010101
10101
101
1

PL/SQL procedure successfully completed.

SQL>
```

d.

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

```
  1  declare
  2  num number:=&n;
  3  i number;
  4  j number;
  5  begin
  6  for i in 1..num
  7  loop
  8     for j in 1..i
  9     loop
 10             DBMS_OUTPUT.PUT(j);
 11     end loop;
 12  DBMS_OUTPUT.NEW_LINE;
 13  end loop;
 14* end;
SQL> /
Enter value for n: 5
old   2: num number:=&n;
new   2: num number:=5;
1
12
123
1234
12345

PL/SQL procedure successfully completed.

SQL>
```

3. Write trigger on employee table for insert, update and delete, make appropriate entries in following table.

Create table emp_check(

Empid number;

Ename varchar2(20),

Oldsal number(9,2),

Newsal number(9,2),

Uname varchar2(20),

Chk_date date);

```
SQL> ED
Wrote file afiedt.buf

  1  CREATE TABLE EMP_CHECK(
  2  EMPNO NUMBER,
  3  ENAME VARCHAR2(20),
  4  OLD_SAL NUMBER,
  5  NEW_SAL NUMBER,
  6  ACTION VARCHAR2(20),
  7  USERNAME VARCHAR2(20),
  8* DATE_CHK DATE)
SQL> /

Table created.

SQL> ED
Wrote file afiedt.buf

  1  CREATE OR REPLACE TRIGGER MONITOR_EMP
  2  AFTER INSERT OR DELETE OR UPDATE ON EMP
  3  FOR EACH ROW
  4  BEGIN
  5  IF INSERTING THEN
  6  INSERT INTO EMP_CHECK VALUES(:NEW.EMPNO,:NEW.ENAME,NULL,:NEW.SAL,'INSERT',USER,SYSDATE);
  7  ELSiF DELETING THEN
  8  insert into emp_check values(:old.empno,:old.ename,:old.sal,null,'delete',user,sysdate);
  9  else
 10  insert into emp_check values(:old.empno,:old.ename,:old.sal,:new.sal,'update',user,sysdate);
 11  end if;
 12* end;
SQL>
SQL> /

Trigger created.
```

```
SQL> insert into emp values(7800,'sssss','CEO',007,sysdate,10000,500,10,1000);

1 row created.

SQL> select * from emp_check;

     EMPNO ENAME                OLD_SAL    NEW_SAL ACTION              USERNAME            DATE_CHK
---------- -------------------- ---------- ---------- -------------------- -------------------- ---------
      7800 sssss                             10000 INSERT              DBDA26              04-OCT-23
```

```
SQL>
SQL> delete from emp where empno=7800;

1 row deleted.

SQL> select * from emp_check;

     EMPNO ENAME                OLD_SAL    NEW_SAL ACTION              USERNAME             DATE_CHK
---------- -------------------- ---------- ---------- -------------------- -------------------- ---------
      7800 sssss                             10000 INSERT              DBDA26               04-OCT-23
      7800 sssss                  10000             delete              DBDA26               04-OCT-23

SQL>
```

```
SQL> update emp set sal =1234 where empno=7900;

1 row updated.

SQL> select * from emp_check;

     EMPNO ENAME                OLD_SAL    NEW_SAL ACTION              USERNAME            DATE_CHK
---------- -------------------- ---------- ---------- -------------------- -------------------- ---------
      7800 sssss                             10000 INSERT              DBDA26              04-OCT-23
      7800 sssss                  10000             delete              DBDA26              04-OCT-23
      7900 JAMES                    950       1234 update              DBDA26              04-OCT-23

SQL>
```