Practical No.1

Q.1] Write a pl/sql block to calculate square of a number by accepting number from user.

```
SQL> declare
  2        n number:=&num;
  3        result number;
  4  begin
  5  result:=n*n;
  6  dbms_output.put_line('The sqaure of a number
'||n||'is'||result);
  7        end;
  8  /
Enter value for num: 4
old   2:       n number:=&num;
new   2:       n number:=4;
PL/SQL procedure successfully completed.
SQL> set serveroutput on;
SQL> /
Enter value for num: 5
old   2:       n number:=&num;
```

```
new   2:       n number:=5;

The sqaure of a number 5is25

PL/SQL procedure successfully completed.
```

Q.2] Write a pl/sql block to display addition, subtraction, multiplication and division of any two numbers entered by user.

```
SQL> declare
  2   n1 number:=&n1;
  3   n2 number:=&n2;
  4   result number;
  5
  6  begin
  7  result:=n1+n2;
  8  dbms_output.put_line('Addition is  '||result);
  9  result:=n1-n2;
 10  dbms_output.put_line('Subtraction is  '||result);
 11  result:=n1/n2;
 12  dbms_output.put_line('Division is  '||result);
 13  result:=n1*n2;
 14  dbms_output.put_line('Multiplication is  '||result);
```

```
 15
 16  end;
 17  /
Enter value for n1: 12
old   2:  n1 number:=&n1;
new   2:  n1 number:=12;
Enter value for n2: 30
old   3:  n2 number:=&n2;
new   3:  n2 number:=30;
Addition is  42
Subtraction is  -18
Division is  .4
Multiplication is  360
PL/SQL procedure successfully completed.
SQL> set serveroutput on;
SQL> /
Enter value for n1: 45
old   2:  n1 number:=&n1;
new   2:  n1 number:=45;
```

Enter value for n2: 56

old 3: n2 number:=&n2;

new 3: n2 number:=56;

Addition is 101

Subtraction is -11

Division is .803571428571428571428571428571428571429

Multiplication is 2520

PL/SQL procedure successfully completed.

Q.3] Write a pl/sql block which displays job location of an employee whose ID is to be accepted by user.

```
SQL> declare
id number:=&id;
v varchar2(20);
begin
Select job into v from emp where empno=id;
dbms_output.put_line(v);
end;
/
```

Enter value for id: 7566

old 2: id number:=&id;

new 2: id number:=7566;

MANAGER

PL/SQL procedure successfully completed.

Q.4] Write a pl/sql block to display employee name and salary of an employee whose ID is to be accepted by user.

```
SQL> declare
  2   id number:=&id;
  3   v1 varchar2(20);
  4   v2 number(20);
  5  begin
  6  Select ename,sal into v1,v2 from emp where empno=id;
  7  dbms_output.put_line(v1||' '||v2);
  8  end;
  9  /
```

Enter value for id: 7566

old 2: id number:=&id;

new 2: id number:=7566;

JONES 2975

PL/SQL procedure successfully completed.

Q.5] Write a pl/sql block to display all the details of department table using unconstrained loop. Dept(deptno,dname,loc)

```
SQL> declare
  2   i number:=10;
  3   dnm varchar2(20);
  4   lo varchar2(20);
  5  begin
  6  loop
  7  Select dname,loc into dnm,lo from dept where deptno=i;
  8  dbms_output.put_line(i||' '||dnm||' '||lo);
  9  i:=i+10;
 10  exit when i=50;
 11  end loop;
 12  end;
 13  /
10 ACCOUNTING NEW YORK
20 RESEARCH DALLAS
30 SALES CHICAGO
```

40 OPERATIONS BOSTON

PL/SQL procedure successfully completed.

Practical No.2

Q.1] Create a sequence which will start with 301 and ends up at 350 with interval 4.

```
SQL> Create sequence S1
  2  start with 301
  3  increment by 4
  4  nominvalue
  5  maxvalue 350
  6  /
```

Sequence created.

Q.2] Create a sequence which will start with 51 and ends up with 200 with an interval of 2 let this sequence be ordered on different machine and cycle. Store 5 numbers into cache memory as well.

```
SQL> Create sequence S2
  2  start with 51
  3  increment by 2
  4  nominvalue
  5  maxvalue 200
  6  cycle
```

```
  7  noorder
  8  /
```

Sequence created.

Q.3] Alter a sequence created in question number by making it cycled sequence.

```
SQL> alter sequence S2
  2  CYCLE
  3  /
```

Sequence altered.

Q.4] Drop a sequence that you have created in previous question.

```
SQL> DROP SEQUENCE S1
  2  /
```

Sequence dropped.

Q.5] Insert a sequence into table and generate its value.

```
SQL> insert into emp(empno)
  2  values(S2.nextval);
```

1 row created.

Practical No.3

Q.1] Create a pl/sql block which displays an appropriate message based on the no of rooms of a room of a particular hotel.

```
SQL> Create table HotelA
  2  (roomid number(3),
  3  no_of_rooms number(3),
  4  floor number(3));
Table created.

SQL> Insert into HotelA
  2  values(100,1,1);
1 row created.

SQL> Insert into HotelA
  2  values(101,2,1);
1 row created.

SQL> Insert into HotelA
  2  values(102,2,2);
1 row created.
```

```
SQL> Insert into HotelA
  2  values(103,2,2);
1 row created.

SQL> Insert into HotelA
  2  values(104,3,3);
1 row created.

SQL> Select * from HotelA;
   ROOMID NO_OF_ROOMS      FLOOR
---------- ---------- ----------
       100          1          1
       101          2          1
       102          2          2
       103          2          2
       104          3          3

SQL> declare
```

```
 2   rid number:=&roomid;
 3   nor number;
 4  begin
 5  Select no_of_rooms into nor from HotelA where roomid=rid;
 6  if nor=1 then
 7  dbms_output.put_line('Small');
 8   elsif nor=2 then
 9  dbms_output.put_line('Fairly big');
10  elsif nor=3 then
11  dbms_output.put_line('Lots of room');
12  end if;
13  end;
14  /
Enter value for roomid: 100
old   2:  rid number:=&roomid;
new   2:  rid number:=100;
PL/SQL procedure successfully completed.
SQL> set serveroutput on;
SQL> /
```

```
Enter value for roomid: 102

old   2:  rid number:=&roomid;

new   2:  rid number:=102;

Fairly big

PL/SQL procedure successfully completed.
```

2.] Create a pl/sql block which accepts a number from the user for an arithmetic operation. Perform that arithmetic operation and display the result. Accept two numbers from the user.

```
 1  declare

 2  n1 number:=&n1;

 3  n2 number:=&n2;

 4  o number:=&operator;

 5  begin

 6  dbms_output.put_line('Select your operation: ');

 7  dbms_output.put_line('1.] Addition: ');

 8  dbms_output.put_line('2.] Subtraction: ');

 9  dbms_output.put_line('3.] Multiplication: ');

10  dbms_output.put_line('4.] Division: ');

11  if o=1 then
```

```
 12  dbms_output.put_line('Addition is '||n1+n2);
 13  elsif o=2 then
 14  dbms_output.put_line('Subtraction is '||n1-n2);
 15  elsif o=3 then
 16  dbms_output.put_line('Multiplication is '||n1*n2);
 17  else
 18  dbms_output.put_line('Division is '||n1/n2);
 19  end if;
 20* end;
SQL> /
Enter value for n1: 1
old   2: n1 number:=&n1;
new   2: n1 number:=1;
Enter value for n2: 2
old   3: n2 number:=&n2;
new   3: n2 number:=2;
Enter value for operator: 3
old   4: o number:=&operator;
new   4: o number:=3;
```

Select your operation:

1.] Addition:

2.] Subtraction:

3.] Multiplication:

4.] Division:

Multiplication is 2

PL/SQL procedure successfully completed.

3.] Write a pl/sql block to display message for the grade obtained.

```
declare
  2  grade char(1):='&grade';
  3  appraisal varchar2(20);
  4  begin
  5       CASE grade
  6         when 'A' then
  7    dbms_output.put_line('Excellent');
  8         when 'B' then
  9    dbms_output.put_line('Very Good');
 10         when 'C' then
```

```
11    dbms_output.put_line('Good');
12        when 'D' then
13    dbms_output.put_line('Fair');
14        when 'E' then
15    dbms_output.put_line('Poor');
16        Else
17    dbms_output.put_line('No such grade');
18       end case;
19* end;
20  /
Enter value for grade: A
old   2: grade char(1):='&grade';
new   2: grade char(1):='A';
Excellent
PL/SQL procedure successfully completed.
```

4.] Create a pl/sql block which accepts a number from the user for an arithmetic operation. Perform that arithmetic operation and display the result. Accept two numbers from the user. Using case expression.

```
SQL> declare
  2  n1 number:=&n1;
  3  n2 number:=&n2;
  4  o number:=&operator;
  5  begin
  6  case o
  7  when 1 then
  8  dbms_output.put_line('Addition is  '||n1+n2);
  9  when 2 then
 10  dbms_output.put_line('Subtraction is  '||n1-n2);
 11  when 3 then
 12  dbms_output.put_line('Multiplication is  '||n1*n2);
 13  when 4 then
 14  dbms_output.put_line('Division is '||n1/n2);
 15  end case;
 16  end;
 17  /
Enter value for n1: 1
old   2: n1 number:=&n1;
```

```
new   2: n1 number:=1;

Enter value for n2: 2

old   3: n2 number:=&n2;

new   3: n2 number:=2;

Enter value for operator: 3

old   4: o number:=&operator;

new   4: o number:=3;

Multiplication is  2

PL/SQL procedure successfully completed.
```

Practical No.4

Q.1] Write a pl/sql block to display the table of three using unconstrained loop.

```
SQL> declare
  2   i number:=3;
  3   j number:=1;
  4  begin
  5  loop
  6  dbms_output.put_line(i*j);
  7  j:=j+1;
  8  exit when j=11;
  9  end loop;
 10   end;
 11  /
3
6
9
12
15
18
```

```
21

24

27

30
```

PL/SQL procedure successfully completed.

Q.2] Write a pl/sql block to display all details of department table using while loop.

Dept(deptno,dname,loc)

```
SQL> declare
  2  i number:=10;
  3  dnm varchar2(20);
  4  l varchar2(20);
  5  begin
  6  while i<50
  7  loop
  8  Select dname,loc into dnm,l from dept where deptno=i;
  9  dbms_output.put_line(i||' '||dnm||' '||l);
 10  i:=i+10;
 11  end loop;
```

```
 12  end;
 13  /
10 ACCOUNTING NEW YORK
20 RESEARCH DALLAS
30 SALES CHICAGO
40 OPERATIONS BOSTON
PL/SQL procedure successfully completed.
```

Q.3] Write a pl/sql block to display all details of department table using for loop.

```
SQL> declare
  2  j number:=10;
  3  i number;
  4  dnm varchar2(20);
  5  l varchar2(20);
  6  begin
  7  for i in 1..4
  8  loop
  9  Select dname,loc into dnm,l  from dept where deptno=j;
 10  dbms_output.put_line(i||'  '||dnm||'  '||l);
```

```
 11  j:=j+10;
 12  end loop;
 13  end;
 14  /
1  ACCOUNTING  NEW YORK
2  RESEARCH  DALLAS
3  SALES  CHICAGO
4  OPERATIONS  BOSTON
```

PL/SQL procedure successfully completed.

Practical 5

Q.1] Demonstrate the use of goto statement. Write a pl/sql block which will print the department name and number for all records except operations department for which it will print all the details of the table.

```
SQL> declare
 2  i number:=1;
 3  j number:=10;
 4  dn varchar2(20);
 5  l varchar2(20);
 6  begin
 7  while i<=4
 8  loop
 9  Select dname into dn from dept where deptno=j;
10  if dn='OPERATIONS' then
11  goto  l1;
12  else
13  dbms_output.put_line(j||'  '||dn);
14  end if;
15  j:=j+10;
```

```
 16  end loop;
 17  <<l1>>
 18  Select loc into l from dept where deptno=j;
 19  dbms_output.put_line(j||'  '||dn||'  '||l);
 20  end;
 21  /
PL/SQL procedure successfully completed.
SQL> set serveroutput on;
SQL> /
10  ACCOUNTING
20  RESEARCH
30  SALES
40  OPERATIONS  BOSTON
PL/SQL procedure successfully completed.
```

Q.2.] Demonstrate the use of null statement.

```
SQL> declare
  2  i number:=10;
  3  dnm varchar2(20);
```

```
 4  l varchar2(20);
 5  begin
 6  loop
 7  Select dname,loc into dnm,l from dept where deptno=i;
 8  dbms_output.put_line(i||'  '||dnm||'  '||l);
 9  i:=i+10;
10  null;
11  exit when i=50;
12  end loop;
13  end;
14  /
10  ACCOUNTING  NEW YORK
20  RESEARCH  DALLAS
30  SALES  CHICAGO
40  OPERATIONS  BOSTON
PL/SQL procedure successfully completed.
```

Practical No. 6

Q.1] Create a procedure to calculate square of a number using (in out) parameter.

CODE:-

```
 1  create or replace procedure P3(n in out number)
 2  is
 3  begin
 4  n:=n*n;
 5* end;
 6  /


Procedure created.

SQL> set serveroutput on;
SQL> declare
 2  num number:=&num;
 3  begin
 4  P3(num);
 5  dbms_output.put_line(num);
 6  end;
 7  /
Enter value for num: 4
old   2: num number:=&num;
new   2: num number:=4;
```

PL/SQL procedure successfully completed.

Q2] Create a procedure to perform addition, subtraction, multiplication and division using IN parameter.

CODE:-

```
 1  create or replace procedure P1(p in number, q in number)
 2  is
 3  addition number;
 4  subtraction number;
 5  multiplication number;
 6  division number;
 7  begin
 8  addition:=p+q;
 9  dbms_output.put_line('addition of numbers ' ||p|| ' and ' ||q|| ' is ' ||addition);
10  subtraction:=p-q;
11  dbms_output.put_line('subtraction of numbers ' ||p||'and'||q||'is'|| subtraction);
12  multiplication:=p*q;
13  dbms_output.put_line('Multiplication of numbers ' ||p|| ' and ' ||q|| ' is ' || multiplication)
14  division:=p/q;
15  dbms_output.put_line(' Division of number ' ||p|| ' and ' ||q|| ' is ' || division);
16* end;
17  /
```

Procedure created.

SQL> execute P1(4,8);

addition of numbers 4 and 8 is 12

subtraction of numbers 4and8is-4

Multiplication of numbers 4 and 8 is 32

Division of number 4 and 8 is .5

PL/SQL procedure successfully completed.

OR

```
SQL> create or replace procedure P1(p in number, q in number)
  2  is
  3  add number;
  4  sub number;
  5  mul number;
  6  div number;
  7  begin
  8  add:=p+q;
  9  dbms_output.put_line('Addition of number '||p||' and '||q||' is '||add);
 10  sub:=p-q;
 11  dbms_output.put_line('Subtraction of number '||p||' and '||q||' is '||sub);
 12  mul:=p*q;
 13  dbms_output.put_line('Multiplication of number '||p||' and '||q||' is '||mul);
```

```
14  div:=p/q;
15  dbms_output.put_line('Division of number '||p||' and '||q||' is '||div);
16  end;
17  /
```

Procedure created.

SQL> exec P1(2,5);

Addition of number 2 and 5 is 7

Subtraction of number 2 and 5 is -3

Multiplication of number 2 and 5 is 10

Division of number 2 and 5 is .4

PL/SQL procedure successfully completed.

Q3] Create a procedure to check whether the given number is even or odd.

CODE:-

```
1  create or replace procedure S5(n in number,result out varchar2)
2  is
3  begin
4  if mod(n,2)=0 then
5  result:='The number '||n||'is even';
```

```
 6  else
 7  result:='The number ' ||n|| 'is odd';
 8  end if;
 9* end;
10  /
```

Procedure created.

```
SQL> execute S5(5,:V);
```

PL/SQL procedure successfully completed.

```
SQL> print V;

V
------------------------------
```

The number 5 is odd

Practical 7

Q 1] Find cube of number using function.

CODE:-

```
SQL> create function f1(n in number)
  2  return number
  3  is
  4  a number;
  5  begin
  6  a:=n*n*n;
  7  return a;
  8  end;
  9  /

Function created.

  1  declare
  2  n number:=&n;
  3  result number;
  4  begin
  5  result:=f1(n);
  6  dbms_output.put_line(result);
  7* end;
  8  /
Enter value for n: 5
```

old 2: n number:=&n;

new 2: n number:=5;

125

PL/SQL procedure successfully completed.

Q.2] Create a function and execute it using dual.

CODE:-

```
SQL> create function f(n in number)
 2  return number
 3  is
 4  begin
 5  if n=1 then return 1;
 6  else
 7  return n*f(n-1);
 8  end if;
 9  end;
10  /
```

Function created.

```
SQL> select f(5) from dual;

      F(5)
----------
       120
```

```
SQL> execute dbms_output.put_line(f(4));
24

PL/SQL procedure successfully completed.
```

Q.3] Calculate factorial of a number using recursive function.

```
SQL> Create function f(n in number)
  2  return number
  3  is
  4  begin
  5  if n=1 then return 1;
  6  else
  7  return n*f(n-1);
  8  end if;
  9  end;
 10  /

Function created.

SQL> exec dbms_output.put_line(f(5));
120

PL/SQL procedure successfully completed.
```

Practical No.8

Q.1] Display the details of department table using cursor.

Code:

```
SQL> declare
  2  cursor c1 is Select*from dept;
  3  record dept%ROWTYPE;
  4  begin
  5  open c1;
  6  loop
  7  fetch c1 into record;
  8  exit when c1%NOTFOUND;
  9  dbms_output.put_line(record.deptno||' '||record.loc);
 10  end loop;
 11  close c1;
 12  end;
 13  /

PL/SQL procedure successfully completed.

SQL> set serveroutput on;
```

```
SQL> /

10 NEW YORK

20 DALLAS

30 CHICAGO

40 BOSTON


PL/SQL procedure successfully completed.
```

Q.2]Write a pl/sql block to delete dname from dept table where deptno is 40 or 7788.

Code:

```
SQL> declare
  2  name varchar(20);
  3  begin
  4  Select dname into name from dept where deptno=40;
  5  loop
  6  if sql%found then delete from dept where deptno=40;
  7  else exit;
  8  end if;
  9  end loop;
 10  end;
```

```
11 /

PL/SQL procedure successfully completed.

SQL> Select*from dept;

    DEPTNO DNAME         LOC
---------- -------------- -------------
        10 ACCOUNTING     NEW YORK
        20 RESEARCH       DALLAS
        30 SALES          CHICAGO
OR
SQL> declare
  2  name varchar(20);
  3  begin
  4  Select ename into name from emp where empno=7788;
  5  if sql%found then delete from emp where empno=7788;
  6  end if;
  7  end;
  8  /

PL/SQL procedure successfully completed.
```

SQL> Select*from emp;

```
    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


    EMPNO ENAME      JOB              MGR HIREDATE         SAL       COMM
---------- ---------- --------- ---------- --------- ---------- ----------
    DEPTNO
---------
      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

EMPNO ENAME JOB MGR HIREDATE SAL COMM

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

DEPTNO

7782 CLARK MANAGER 7839 09-JUN-81 2450

10

7839 KING PRESIDENT 17-NOV-81 5000

10

7876 ADAMS CLERK 7788 23-MAY-87 1100

20

EMPNO ENAME JOB MGR HIREDATE SAL COMM

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

```
   DEPTNO
---------
     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

12 rows selected.
```

Q.3] Write a pl/sql block to delete a record having salary as 1500 from emp table.

Code:

```
SQL> declare
 2  begin
 3  delete from emp where sal=1500;
 4  dbms_output.put_line(sql%rowcount);
 5  end;
 6  /
```

PL/SQL procedure successfully completed.