PL/SQL ASSIGNMENT -1

1. Display your name 5 times using for loop.

Query:

```
DECLARE
name VARCHAR2 (10):= 'Suvodeep';
BEGIN
FOR i IN 1..5
LOOP
DBMS_OUTPUT.put_line (name);
END LOOP;
END;
/
```

Output:

```
Statement processed.
Suvodeep
Suvodeep
Suvodeep
Suvodeep
Suvodeep
```

2. Write A PL/SQL block of code to invert a number 12345 to 54321

Query:

```
DECLARE
num number;
reverse_num number:=0;
begin
num:=98765;
while num>0
loop
reverse_num:=(reverse_num*10) + mod(num,10);
num:=trunc(num/10);
end loop;
dbms_output.put_line(' Reversed number is : '|| reverse_num);
end;
/
```

Output:

```
Statement processed.
 Reversed number is : 54321
```

3. Write a pl/sql code block to calculate the area of circle for a value of radius varying from 3 to 7. store the radius & the corresponding values of calculated area in an empty table named areas, consisting of two columns, radius & area. Hint: create a table Area(radius,area), insert values into Area table through PL/SQL blocks.

Query:

```
create table areas ( r number(2), area number (14,2));
declare r number(5);
area number(14,2);
pi constant number (4,2):=3.14;
begin
r:=3;
while r<=7
loop
area:=pi*power(r,2);
insert into areas values(r,area );
r:=r+1;
end loop;
end;
/
select * from areas;
```

Output:

```
Table created.

Statement processed.
```

R	AREA
3	28.26
4	50.24
5	78.5
6	113.04
7	153.86

Download CSV

5 rows selected.

4. Create a simple loop such that a message is displayed when a loop exceeds a particular value(while loop).

Query:

```
Declare
a number :=1;
BEGIN
dbms_output.put_line('Program started');
LOOP
dbms_output.put_line(a);
a:=a+1;
```

```
Exit when a>5;
end loop;
dbms_output.put_line('Exceeded limit');
end;
  /
```

Output:

```
Statement processed.
Program started
1
2
3
4
5
Exceeded limit
```

6. Write a function to find factorial of a number. Also call and check the factorial of a number.

Query:

```
factorial number;

create or replace FUNCTION fact(a number)

RETURN number

IS

f number;

BEGIN

IF a=0 THEN

f := 1;

ELSE

f := a * fact(a-1);

END IF;

RETURN f;

END ;

/

--calling function

set serveroutput on;
```

```
Declare
a number;
factorial number;
BEGIN
a:= 6;
factorial := fact(a);
dbms_output.put_line(' Factorial'|| a || 'is '|| factorial);
END;
/
```

Output:

```
Statement processed.
 Factorial6is 720
```

7. Write a function to find maximum between two number

Query:

```
DECLARE
a number;
b number;
c number;
FUNCTION findMax(x IN number, y IN number)
RETURN number
IS
z number;
BEGIN
IF x > y THEN
z:= x;
ELSE
Z:= y;
END IF;
RETURN z;
```

```
END;


BEGIN
a:= 23;
b:= 45;


c := findMax(a, b);
dbms_output.put_line(' Maximum of (23,45):' || c);
END;
/
```

Output:

```
Statement processed.
 Maximum of (23,45):45
```

8. Write a Procedure using a IN and an OUT Parameter - to find maximum between two number.

Query:

```
DECLARE
 a number;
 b number;
 c number;
 PROCEDURE findMin(x IN number, y IN number, z OUT number) IS
BEGIN
 IF x < y THEN
 z:= x;
 ELSE
 z:= y;
 END IF;
END;
BEGIN
```

```
a:= 23;
b:= 45;
findMin(a, b, c);
dbms_output.put_line(' Minimum of (23, 45) : ' || c);
END;
/
```

Output:

```
Statement processed.
 Minimum of (23, 45) : 23
```

9. Write a Procedure using IN OUT Parameter – to find cube a number. Also call the procedure to give output for a number.

Query:

```
set SERVEROUTPUT ON;
create or replace procedure INOUTPARA(p IN OUT number)
AS
BEGIN
p:=p*3;
END INOUTPARA;
--calling
declare
x number;
begin
x:= 15;
INOUTPARA(x);
dbms_output.put_line('The result is '||x);
end;
```

Output:

```
Statement processed.
The result is 45
```