

a. Write a PL/SQL code, EX_INVNO.SQL, block for inverting a number using all forms of loops.

```
declare
  n number;
  i number;
  rev number:=0;
  r number;

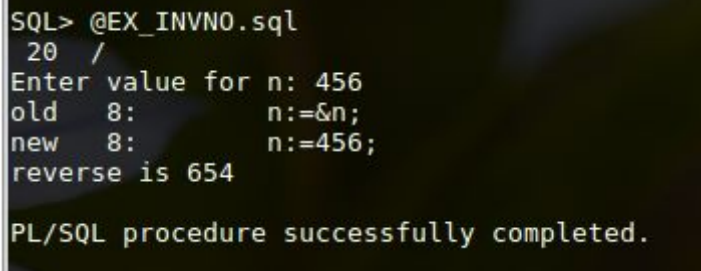
begin
  n:=&n;

  while n>0
  loop
    r:=mod(n,10);
    rev:=(rev*10)+r;
    n:=trunc(n/10);
  end loop;

  dbms_output.put_line('reverse is '||rev);

End;
```

Output:



```
SQL> @EX_INVNO.sql
20 /
Enter value for n: 456
old 8:          n:=&n;
new 8:          n:=456;
reverse is 654

PL/SQL procedure successfully completed.
```

B. Write a PL/SQL code, EX_SUMNO.SQL that prints the sum of 'n' natural numbers.

```
declare
  n number;
  i number:=1;
  s number:=0;
  r number;

begin
  n:=&n;

  while i<n
  loop
```

```

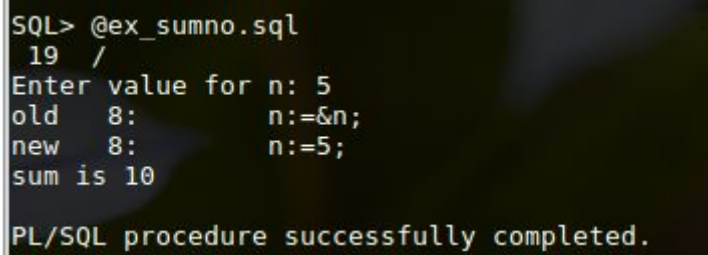
        s:=s+i;
        i:=i+1;
    end loop;

    dbms_output.put_line('sum is '||s);

```

End;

Output:



```

SQL> @ex_sumno.sql
19 /
Enter value for n: 5
old 8:          n:=&n;
new 8:          n:=5;
sum is 10

PL/SQL procedure successfully completed.

```

C. Write a PL/SQL program to print all the prime numbers between 100 and 400

```

DECLARE
    i NUMBER;
    j NUMBER;
BEGIN
    dbms_output.Put_line('The prime numbers are:');
    dbms_output.new_line;
    i := 100;
    LOOP
        j := 2;
        LOOP
            EXIT WHEN( ( MOD(i, j) = 0 )
                OR ( j = i ) );
            j := j + 1;
        END LOOP;
        IF( j = i )THEN
            dbms_output.Put(i||' ');
        END IF;
        i := i + 1;
        exit WHEN i = 400;
    END LOOP;
    dbms_output.new_line;
END;

```

Output

```

SQL> @prime2.sql
23 /
The prime numbers are:
101  103  107  109  113  127  131  137  139  149  151  157  163
167  173  179  181  191  193  197  199  211  223  227  229  233
239  241  251  257  263  269  271  277  281  283  293  307  311
313  317  331  337  347  349  353  359  367  373  379  383  389
397

PL/SQL procedure successfully completed.

```

D. Write a PL/SQL program to print n terms of fibonacci series.

declare

f number := 0;
s number := 1;
temp number;
n number;
i number;

begin

n:=&n;

dbms_output.put_line('Series:');

dbms_output.put_line(f);
dbms_output.put_line(s);

for i in 2..n
loop
temp:=f+s;

f := s;
s := temp;

dbms_output.put_line(temp);
end loop;

end;

Output

```

SQL> @fibo.sql
29 /
Enter value for n: 5
old 10:          n:=&n;
new 10:          n:=5;
Series:
0
1
1
2
3
5

PL/SQL procedure successfully completed.

```

E. Write a PL/SQL program to calculate HCF of two numbers.

```

declare
    u number;
    v number;
    r number := 0;
begin
    u:=&u;
    v:=&v;
    while v != 0
    loop
        r := mod(u,v);
        u := v;
        v := r;
    end loop;
    dbms_output.put_line('HCF = ' || u);
end;

```

OUTPUT

```

SQL> @gcd.sql
16 /
Enter value for u: 16
old 6:      u:=&u;
new 6:      u:=16;
Enter value for v: 12
old 7:      v:=&v;
new 7:      v:=12;
HCF = 4

PL/SQL procedure successfully completed.

```

Write a PL/SQL code, EX_AREA.SQL, of block to calculate the area of the circle for the values of radius varying from 3 to 7. Store the radius and the corresponding values of calculated area in the table AREA_VALUES.

```

declare
    a number := 0.00;

```

```
begin
    for i in 3 .. 7
    loop
        a := i * i * 3.14;
        insert into area_values(radius,area) values (i,a);
    end loop;
end;
/
select * from area_values;
```

OUTPUT

```
SQL> @area.sql

PL/SQL procedure successfully completed.
```

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