```
//reverse no function procedure
CREATE OR REPLACE FUNCTION ReverseNumber(p_num IN NUMBER) RETURN NUMBER IS
v_num NUMBER := p_num;
v_reversed_num NUMBER := 0;
v_remainder NUMBER;
BEGIN
WHILE v_num > 0 LOOP
v_remainder := MOD(v_num, 10); -- Get the rightmost digit
v_reversed_num := (v_reversed_num * 10) + v_remainder; -- Append the digit to the reversed number
v_num := FLOOR(v_num / 10); -- Remove the rightmost digit from the original number
END LOOP;

RETURN v_reversed_num;
END;
/
DECLARE
v_input_num NUMBER := 5639;
v_reversed_num NUMBER;
BEGIN
v_reversed_num := ReverseNumber(v_input_num);
DBMS_OUTPUT.PUT_LINE('Reversed number: ' || v_reversed_num);
EXCEPTION
WHEN OTHERS THEN
DBMS_OUTPUT.PUT_LINE('Error occurred: ' || SQLERRM);
END;
/
//Stored Procedure
SET SERVEROUTPUT ON;
DECLARE
n NUMBER;
rev NUMBER;
r number;
a number;
BEGIN
n:=&n;
rev:=0;
a:=n;
WHILE a>0
LOOP
r:=mod(a,10);
rev:=(rev*10) + r;
a:=floor(a/10);
END LOOP;
DBMS_OUTPUT.PUT_LINE('Reverse of the number is: ' || rev);
END;
/
FIBONACCI
 set serveroutput on
declare
a number:= 0;
b number:= 1;
c number;
n number;
i number;
begin
n:=&n;
dbms_output.put_line('Series:');
dbms_output.put_line(a);
dbms_output.put_line(b);
for i in 2..n
loop
c:=a+b;
a:=b;
b:=c;
dbms_output.put_line(c);
end loop;
end;
/
```

CURSOR PROG

```
set serveroutput on

declare
cursor cur2 is select mv_no,price from movie;
movie_id movie.mv_no%type;
mv_price movie.price%type;
begin
open cur2;
if(cur2%isopen)then
loop
fetch cur2 into movie_id,mv_price;
exit when cur2%notfound;
update movie set price=mv_price*1.02 where mv_no=movie_id;
dbms_output.put_line('Price updated for '|| movie_id);
end loop;
else
dbms_output.put_line('Cursoe is not open');
end if;
end;
/
```

PRIME PROG

```
set serveroutput on
create or replace function test_prime(num number) return number
is

i number;
flag number:=0;
n number;
begin
n:=trunc(num/2);
for i in 2..n
loop
if(mod(num,i)=0)then
flag:=1;
exit;
end if;
end loop;
if(flag=1)then
return 0;
else
return 1;
end if;
return 1;
end;
/
```

```
declare
n number;
x number;
j number:=2;
begin
n:=&n;
while n>0
loop
x:=test_prime(j);
if(x=1)then
dbms_output.put_line(j);
n:=n-1;
end if;
j:=j+1;

end loop;
end;
/
```

VIEW

```
create view vw_invoice_master as select
s1.inv_no,s1.mv_no,s2.title,s2.type,s2.price,s1.cust_id,s3.fname,s3.lname from invoice s1,customer s3,movie s2 where s1.cust_id=s3.cust_id and s1.mv_no=s2.mv_no;
update vw_invoice_master set price=500 where inv_no='I03';
delete from vw_invoice_master where mv_no='M001';
drop view vw_invoice_master;
```