1.

```
CREATE OR REPLACE FUNCTION add_two_nums(num1 NUMERIC,num2 NUMERIC)
RETURNS NUMERIC AS'
BEGIN
   RETURN num1 + num2;
END;
'LANGUAGE PLPGSQL;
-- create

SELECT add_two_nums(5,6);
```

2.

```
create or replace function max_no (in A int, in B int) returns int as'
declare
begin
if A>B then
raise notice '' first number = % is greater '',A;
return A;
else
raise notice '' second number = % is greater '', B;
return B;
end if;
end;
'language 'plpgsql';
select max_no(4,3);
```

3.

```
create or replace function min_no (in A int, in B int) returns int as'
declare
begin
if A<B then
raise notice '' first number = % is minimum '',A;
```

```
return A;
else
raise notice '' second number = % is minimum'', B;
return B;
end if;
end;
'language 'plpgsql';
select min_no(6,9);
```

4.

```
create or replace function num (in A int) returns int as'
declare
begin
if A>0 then
raise notice '' number is = % is positive '' , A;
return A;
else if A<0 then
raise notice '' number is = % is negative '', A;
return A;
else if A=0 then
raise notice '' number is = % is zero '', A;
return A;
end if;
end if;
end if;
end;
'language 'plpgsql';
select num(4);
```

5.

```
create or replace function max_num(in A int, in B int, in C int ) returns int as'
declare
begin
if A>B and A>C then
raise notice '' first number = % is maximum'',A;
return A;
else if B>A and B>C then
raise notice '' second number = % is maximum '' ,B;
return B;
else
raise notice '' third number = % is maximum '' ,C;
return C;
end if;
end if;
end;
'language 'plpgsql';
select max_num(1,5,4);
```

6.

```
create or replace function min_num(in A int, in B int, in C int ) returns int as'
declare
begin
if A<B and A<C then
raise notice '' first number = % is minimum'',A;
return A;
else if B<A and B<C then
raise notice '' second number = % is minimum '' ,B;
return B;
else
raise notice '' third number = % is minimum'' ,C;
```

```
return C;
end if;
end if;
end;
'language 'plpgsql';
select min_num(45,6,20);
```

7.

```
create or replace function even_odd (in A int) returns int as'
declare
begin
if A%2=0 then
raise notice '' first no is =% even '',A;
return A;
else
raise notice '' second no is =% odd '',A;
return A;
end if ;
end;
'language 'plpgsql';
select even_odd(5);
```