9 Write a procedure P_LARGE that prints the largest number of given number. (N1 = 10, N2 = 20, N3 = 30). (Use nested if)

→
```
Drop procedure if exists P_Large;
Delimiter //
create procedure P_Large()
Begin
        Declare n1, n2, n3 int;
        Set n1 = 10, n2 = 20, n3 = 30;

    if (n1 > n2) then
        if (n1 > n3) then
            Select 'N1 is greater';
        else
            Select 'N3 is greater';
        end if;
    else
        if (n2 > n3) then
            Select 'N2 is greater';
        else
            Select 'N3 is greater';
        end if;
    end if;
End //
Delimiter ;
Call P_Large();
```

→

N3 is greater
N3 is greater

10 Write a procedure P_CITY that will display the name and total rating of the city that has highest total rating.

→
```
Drop procedure if exists P_CITY;
Delimiter //
Create procedure P_CITY()
Begin
        Declare VCITY varchar(20);
        Declare VRAT int;

        select CITY, sum(RATING) as SumR into VCITY, VRAT
        from customer group by CITY order by SumR
        desc limit 1;

        select VCITY, VRAT;
End//
Delimiter ;
call P_CITY();
```

→

VCITY	VRAT
Surat	500

11 Write a procedure P_GRADE that will display the grade of the CITY that has highest total rating. The grade of the CITY will be decided according to following rules.
1. If total rating of all customer of that city is less than 1000 then CITY_GRADE will be 'POOR'.
2. If total rating of all customer of that city is more than or equal to 1000 and less than 2000 then CITY_GRADE will be 'GOOD'.
3. If total rating of all customer of that city is more than or equal to 2000 and less than 3000 then CITY_GRADE will be 'EXCELLENT'.
4. If total rating of all customer of that city is more than or equal to 3000 then CITY_GRADE will be 'OUTSTANDING'. Use elseif structure.

→
```
Drop procedure if exists P_GRADE;
Delimiter //
create procedure P_GRADE()
Begin
    Declare VCITY varchar(20);
    Declare VRAT int;

    select CITY, sum(RATING) as SumR into VCITY, VRAT
    from customer group by CITY order by SumR
    desc limit 1;

    if (VRAT<1000) then
        select 'POOR' as Grade;
    elseif (VRAT>=1000 & VRAT<20000) then
        select 'GOOD' as Grade;
```

```
        elseif (VRAT>=2000 & VRAT<3000) then
            Select 'EXCELLENT' as Grade;
        else
            Select 'OUTSTANDING' as Grade;
        end if;
End//
Delimiter ;
Call PGRADE();
```

→

Grade
POOR

12 Rewrite the procedure Created in 3.4 using nested if.

→
```
Drop procedure if exists P_GRADE;
Delimiter //
Create procedure P_GRADE()
Begin
    Declare VCITY varchar(20);
    Declare VRAT int;

    Select CITY, sum(RATING) as SumR int VCITY, VRAT
    from customer group by CITY order by SumR
    desc limit 1;

    If (VRAT<1000) then
        Select 'POOR' as Grade;
    End if;

    If (VRAT>=1000) then
        If (VRAT<2000) then
            Select 'GOOD' as Grade;
        End if;
    End if;

    If (VRAT>=2000) then
        If(VRAT<3000) then
            Select 'EXCELLENT' as Grade;
        else
            Select 'OUTSTANDING' as Grade;
        End if;
    End if;
```

End//
Delimiter ;
Call P_GRADE();

→

Grade
POOR

13 Write a procedure P_GRADE that will print customer name and grade of the customer whose customer number is 2002.

Grade will be decided according to following rules :

1. If rating is 100 then grade will be 'POOR'.
2. If rating is 200 then grade will be 'GOOD'.
3. If rating is 300 then grade will be 'EXCELLENT'.

Use simple case structure.

→ Drop procedure if exists P_GRADE;

Delimiter //

Create procedure P_GRADE()

Begin

Declare VCNAME ~~CNAME~~ varchar(20);

Declare URAT int;

Select CNAME, sum(CRATING) as SumR into VCNAME, URAT from customer where CNUM = 2002 group by CNAME;

/* Mark starting of CASE block */

CASE

when URAT = 100 then

select 'POOR' as Grade;

when URAT = 200 then

select 'GOOD' as Grade;

when URAT = 300 then

select 'EXCELLENT' as Grade;

/* Mark ending of CASE block */

End ~~cash~~ case;

End //

Delimiter ;

Call P_GRADE();

Grade
GOOD

14 Rewrite the procedure for problem statement 13 using search case structure.

→
```
Drop procedure if exists P_GRADE;
Delimiter //
Create procedure P_GRADE()
Begin
    Declare VCNAME varchar(20);
    Declare VRAT int;

    select CNAME, SUM(RATING) as SUMR into VCNAME, VRAT
    from Customer where CNUM = 2002 group by CNAME;

    Case VRAT
        when 100 then
            Select 'POOR' as Grade;
        when 200 then
            Select 'GOOD' as Grade;
        when 300 then
            Select 'EXCELLENT' as Grade;
    End case;
End //
Delimiter ;
Call P_GRADE();
```

→

Grade
GOOD

15 Write a procedure P_GRADE that will print customer name and grade of the customer whose

16 Rewrite the procedure for problem statement 11 using simple case structure.

→
```
Drop procedure if exists P_GRADE;
Delimiter //
Create procedure P_GRADE()
Begin
    Declare UCITY varchar(20);
    Declare URAT int;

    select CITY, sum(RATING) as SumR into UCITY, URAT
    from customer group by CITY order by SumR
    desc limit 1;

    CASE
        when URAT <1000 then
            select 'POOR' as Grade;
        when URAT >=1000 & URAT < 2000 then
            select 'GOOD' as Grade;
        when URAT >=2000 & URAT < 3000 then
            select 'EXCELLENT' as Grade;
        when URAT >=3000 then
            select 'OUTSTANDING' as Grade;
    End case;
End//
Delimiter ;
Call P_GRADE();
```

Grade
POOR

15 which case structure (Simple or Search) will be appropriate for following procedure ?

Write a procedure P_GRADE that will print customer name and grade of the customer whose customer number is 2002. Grade will be decided according to following rules.

1. If rating is between 0-100 then grade will be 'POOR'.
2. If rating is between 101-200 then grade will be 'GOOD'
3. If rating is between 201-300 then grade will be 'EXCELLENT'.

→ Drop procedure P_GRADE()

```
Begin
    Declare VCNAME varchar(20);
    Declare VRAT int;

    Select CNAME, RATING into VCNAME, VRAT from
    Customer where CNUM = 2002;

    Case
        when VRAT>=0 & VRAT<=100 then
            Select 'POOR' as grade;
        when VRAT>=101 & VRAT<=200 then
            Select 'Good' as Grade;
        when VRAT>=201 & VRAT<=300 then
            Select 'EXCELLENT' as grade;
    End case;
End//
Delimiter ;
Call P_GRADE();
```

Grade
POOR