Some Basic Sample SQL Query

Tables for Question 1 and 2
-- Marks: Stu_ID | Sub_ID | Sub_Name | Class | Marks
-- Student: Stu_ID | Stu_Name

-- **Question:1 Calculate total marks of every student. i.e. summation of marks scored by a student in**
all his/her subjects.

-- Output: Stu_name | Class | Total_marks

```
-- Query:
  --Assuming that all the students name are distinct in each class
select
    Stu_name,
    Class,
    Sum(Marks) as Total_Marks
from
    Student
join
     marks
on
     Student.Stu_ID=Marks.Stu_ID
Group by
     Stu_name,
     class

--Assuming that all the students name are not distinct in each class
select
   Stu_ID,
   Stu_name,
   Class,
   Sum(Marks) as Total_Marks
from
   Student
join
   marks
on
   Student.Stu_ID=Marks.Stu_ID
Group by
   Stu_ID,Stu_name,class
```

-- Question:2 Calculate rank of every student within a class based on their total marks. i.e. within
class 9 who is 1 st, 2 nd and 3 rd . Similarly rank should be available for other classes, class 10, class 11 etc.
-- Output: Stu_name | Class | Total_marks | Rank

-- Query:

```
--Assuming that all the students name are distinct in each class
select
   Stu_name,
   Class,
   Sum(Marks) as Total_Marks,
   Rank() Over(Partition by Class order by Sum(Marks) desc) as Rank
from
   Student
join
   marks
on
   Student.Stu_ID=Marks.Stu_ID
Group by
   1,2,4
```

```
--Assuming that all the students name are not distinct in each class
select
   Stu_ID,
   Stu_name,
   Class,
   Sum(Marks) as Total_Marks,
   Rank() Over(Partition by Class order by Sum(Marks) desc) as Rank
from
   Student
join
   marks
on
   Student.Stu_ID=Marks.Stu_ID
Group by
   1,2,3,5
--Assuming that two students have same rank then rank function will assign same rank to them and skip the next rank marking (next +1) to the next student for example
--student class rank
```

--a 1 1
--b 1 1
--c 1 3
--This issue will be resolved with Dense rank it will maintain the chronological order assigning rank 2 to student c

```
select
   Stu_name,
   Class,
   Sum(Marks) as Total_Marks,
   Dense_Rank() Over(Partition by Class order by Sum(Marks) desc) as Rank
from
   Student
join
   marks
on
   Student.Stu_ID=Marks.Stu_ID
Group by
   1,2,4
```

-- **Question: 3 There is a call centre and they record incoming call time in two separate tables:**
Table 1: start_time
Phone_number | start_time
This table records the incoming call number and the start time of that call.
Table 2: end_time
Phone_number | end_time
This table records the incoming call number and the end time of that call.

Create a query to join these two tables to get the start and end time of the call together and also
calculate the call duration of each call. Please do take into account that there will be multiple calls
from one phone number and each entry in start table has a corresponding entry in end table.

--Query

```
With start_times as (
      select
         Phone_number,
         start_time,
         Row_Number() Over(Partition by Phone_number order by start_time asc) as Rn
      from
         start_time),

end_times as (
```

```
        select
            Phone_number,
            end_time,
            Row_Number() Over(Partition by Phone_number order by end_time asc) as Rn
        from
            end_time)
Select
    st.Phone_number,
    Start_time,
    End_time,
    timediff(end_time,start_time) as duration
from
    start_times st
Join
    end_times et
on
    st.Phone_number=et.Phone_number and st.Rn=et.Rn
```