

Relevel Practice Round 3 - SQL- II

Relevel
by Unacademy



Case Study 1



Case Study 1

There is one Car manufacturing company ABC. It has a dealer sales model where cars are sold through dealerships. There are 5 models that ABC sells

Table -

Dealer_ID	Model_ID	Engine_No	Current Status (Inventory, Transit, Retailed)	Inventory Days (Inventory Days starts calculating when the vehicle reaches the dealership)	Booking_Id
ABC1	123A	XYZ123	Inventory	10	
ABC1	123C	XYZ134	Transit	N/A	123ABC
ABC1	123D	XYZ128	Retailed	21	124ABC
ABC2	123B	XYZ128	Inventory	7	123ADF
ABC2	123E	XYZ129	Transit	N/A	
ABC2	123C	XYZ125	Retailed	10	124XYZ

csv file- https://drive.google.com/file/d/19ljcJ-BPMcoJJsGYHev_bB3oIQTTfzip/view?usp=sharing

Case Study 1

Question 1. We need to understand the behaviour of manufacturing and stocking up of cars to manage the efficiency of booking. We need to improve the profitability of dealership by decreasing the inventory days. For this we can do either of two things- Either sell the cars quickly or keep only the relevant cars in the inventory. Let us start analyzing the data by calculating Model wise Average Inventory days at each dealership by overall model wise average inventory days to see if the model is stocked up for this particular dealer only or generally has high inventory days.

Dealer_ID	Model_ID	Engine_No	Current Status (Inventory, Transit, Retailed)	Inventory Days (Inventory Days starts calculating when the vehicle reaches the dealership)	Booking_Id
ABC1	123A	XYZ123	Inventory	10	
ABC1	123C	XYZ134	Transit	N/A	123ABC
ABC1	123D	XYZ128	Retailed	21	124ABC
ABC2	123B	XYZ128	Inventory	7	123ADF
ABC2	123E	XYZ129	Transit	N/A	
ABC2	123C	XYZ125	Retailed	10	124XYZ

Case Study 1

Solution:

{concepts used : average function,joins}

```
Select case when A.Dealer_Inventory_day>Model_Inventory_Days then 'Check the Dealer'  
          When A.Dealer_Inventory_day<Model_Inventory_Days then 'Check the Model'  
          Else null end as flag
```

```
From (
```

```
(Select avg(Inventory Days) Dealer_Inventory_days,model_id,Dealer_ID
```

```
* from
```

```
Car ) A
```

```
Join
```

```
(select avg(Inventory Days) Model_Inventory_Days,model_id
```

```
* from
```

```
Car) B) base
```



Case Study 1

Question 2. We can also determine the performance of a dealership by calculating Booking and total inventory ratio. (The Cars in Transit are also in inventory). Rank the dealerships on the basis of this ratio to compare the dealership performance. Higher the ratio better is the performance

Dealer_ID	Model_ID	Engine_No	Current Status (Inventory, Transit, Retailed)	Inventory Days (Inventory Days starts calculating when the vehicle reaches the dealership)	Booking_Id
ABC1	123A	XYZ123	Inventory	10	
ABC1	123C	XYZ134	Transit	N/A	123ABC
ABC1	123D	XYZ128	Retailed	21	124ABC
ABC2	123B	XYZ128	Inventory	7	123ADF
ABC2	123E	XYZ129	Transit	N/A	
ABC2	123C	XYZ125	Retailed	10	124XYZ

Case Study 1

Solution:

{concepts used: subqueries, group by statement , windows functions,distinct statement}

```
Select *, row_number() over(order by Ratio desc) as rank_Dealers  
from (
```

```
Select  
(count(distinct(Case when Booking_Id is not null then Booking_Id else null end )) as Booking/  
count(distinct(Case when Current Status in ('Transit','Inventory') then Booking_Id else null end )) as Total_Inventory ) as Ratio,  
Dealer_ID  
* from  
Car  
Group by Dealer_ID) base
```



Case Study 1

Question 3. We also want to look at other performance metrics of a dealer like time taken to retail a particular model. For this we will first make a key called 'Dealer_ID'- 'Model_ID' and create a table called car_v1

Dealer_ID	Model_ID	Engine_No	Current Status (Inventory, Transit, Retailed)	Inventory Days (Inventory Days starts calculating when the vehicle reaches the dealership)	Booking_Id
ABC1	123A	XYZ123	Inventory	10	
ABC1	123C	XYZ134	Transit	N/A	123ABC
ABC1	123D	XYZ128	Retailed	21	124ABC
ABC2	123B	XYZ128	Inventory	7	123ADF
ABC2	123E	XYZ129	Transit	N/A	
ABC2	123C	XYZ125	Retailed	10	124XYZ

Case Study 1

Solution:

{concepts used: concat string function}

```
Create table car_v1 as
Select
*,concat( Dealer_ID , ' - ' , Model_ID) as Key, avg(Inventory days)
from
Car
Where Current Status = 'Retailled'
```



Case Study 1

Question 4. For each dealer we want to calculate the number of days for which it keeps the car. Let's say that it keeps almost all models in inventory for 10 days. So we want this number of days for which most of the cars are kept in the inventory by the dealer to understand the flaw of the system. We wish to create a frequency table where we want to see for a particular dealer and Inventory Days how many cars are stocked by Example dealer_ID = ABC1 for 10 inventory days stocks up 5 cars.

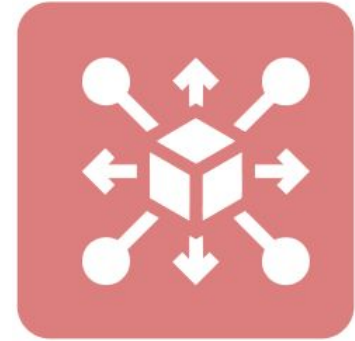
Dealer_ID	Model_ID	Engine_No	Current Status (Inventory, Transit, Retailed)	Inventory Days (Inventory Days starts calculating when the vehicle reaches the dealership)	Booking_Id
ABC1	123A	XYZ123	Inventory	10	
ABC1	123C	XYZ134	Transit	N/A	123ABC
ABC1	123D	XYZ128	Retailed	21	124ABC
ABC2	123B	XYZ128	Inventory	7	123ADF
ABC2	123E	XYZ129	Transit	N/A	
ABC2	123C	XYZ125	Retailed	10	124XYZ

Case Study 1

Solution:

{Concepts used: count function}

```
Select Inventory_days,Dealer_ID, count(Engine_No) as cnt  
from  
Car  
Group by Inventory_days,Dealer_ID
```



Case Study 1

Question 5. We have to send out our new launched models to only those dealers which are popular and have good bookings. We will shortlist it on the basis of minimum number of bookings that is 50 . Write a query to identify the dealer who have less than 50 bookings till date.

Dealer_ID	Model_ID	Engine_No	Current Status (Inventory, Transit, Retailed)	Inventory Days (Inventory Days starts calculating when the vehicle reaches the dealership)	Booking_Id
ABC1	123A	XYZ123	Inventory	10	
ABC1	123C	XYZ134	Transit	N/A	123ABC
ABC1	123D	XYZ128	Retailed	21	124ABC
ABC2	123B	XYZ128	Inventory	7	123ADF
ABC2	123E	XYZ129	Transit	N/A	
ABC2	123C	XYZ125	Retailed	10	124XYZ

Case Study 1

Solution:

{**concepts used: count function**}

```
Select count(Booking_ID) as bookings, dealer_id  
From car  
Group by dealer_id  
Having count(Booking_ID)< 50
```



Case Study 2



Case Study 2

Data Mart is Danny's latest venture and after running international operations for his online supermarket that specializes in fresh produce - Danny is asking for your support to analyze his sales performance.

In June 2020 - large scale supply changes were made at Data Mart. All Data Mart products now use sustainable packaging methods in every single step from the farm all the way to the customer.

Danny needs your help to quantify the impact of this change on the sales performance for Data Mart and its separate business areas.

The key business question he wants you to help him answer are the following:

- What was the quantifiable impact of the changes introduced in June 2020?
- Which platform, region, segment and customer types were the most impacted by this change?
- What can we do about future introduction of similar sustainability updates to the business to minimize impact on sales?

Case Study 2

Example Rows

10 random rows are shown in the table output below from data_mart.weekly_sales:

Week_date	Region	Platform	Segment	Customer_type	Transactions	Sales
9/9/20	OCEANIA	Shopify	C3	New	610	110033.89
29/7/20	AFRICA	Retail	C1	New	110692	3053771.19
22/7/20	EUROPE	Shopify	C4	Existing	24	8101.54
13/5/20	AFRICA	Shopify	null	Guest	5287	1003301.37
24/7/19	ASIA	Retail	C1	New	127342	3151780.41
10/7/19	CANADA	Shopify	F3	New	51	8844.93
26/6/19	OCEANIA	Retail	C3	New	152921	5551385.36
29/5/19	SOUTH AMERICA	Shopify	null	New	53	10056.2
22/8/18	AFRICA	Retail	null	Existing	31721	1718863.58
25/7/18	SOUTH AMERICA	Retail	null	New	2136	8

csv file- <https://drive.google.com/file/d/1aHQ-GYi8XI3DBREiWIVtmju7cZrDdyxH/view?usp=sharing>

Case Study 2

Question 1. Data Cleansing Steps

In a single query, perform the following operations and generate a new table in the data_mart schema named clean_weekly_sales:

- Add a week_number as the second column for each week_date value, for example any value from the 1st of January to 7th of January will be 1, 8th to 14th will be 2 etc
- Add a month_number with the calendar month for each week_date value as the 3rd column
- Add a calendar_year column as the 4th column containing either 2018, 2019 or 2020 values

Week_date	Region	Platform	Segment	Customer_type	Transactions	Sales
9/9/20	OCEANIA	Shopify	C3	New	610	110033.89
29/7/20	AFRICA	Retail	C1	New	110692	3053771.19
22/7/20	EUROPE	Shopify	C4	Existing	24	8101.54
13/5/20	AFRICA	Shopify	null	Guest	5287	1003301.37
24/7/19	ASIA	Retail	C1	New	127342	3151780.41
10/7/19	CANADA	Shopify	F3	New	51	8844.93
26/6/19	OCEANIA	Retail	C3	New	152921	5551385.36
29/5/19	SOUTH AMERICA	Shopify	null	New	53	10056.2
22/8/18	AFRICA	Retail	null	Existing	31721	1718963.58
25/7/18	SOUTH AMERICA	Retail	null	New	2136	8

Case Study 2

Solution:

{Concept used : date time functions and create table}

```
Create table clean_weekly_sales as
Select * , week(week_date) as week_number,month(week_date) as month_number,year(week_date) as calander_year
from
    Data_mart.weekly_sales
```



Case Study 2

Question 2. Add a new column called age_band after the original segment column using the following mapping on the number inside the segment value

Note: Here you can use (optional) newly generated table from question 1.

- Add a new demographic column using the following mapping for the first letter in the segment values:

Week_date	Region	Platform	Segment	Customer_type	Transactions	Sales
9/9/20	OCEANIA	Shopify	C3	New	610	110033.89
29/7/20	AFRICA	Retail	C1	New	110692	3053771.19
22/7/20	EUROPE	Shopify	C4	Existing	24	8101.54
13/5/20	AFRICA	Shopify	null	Guest	5287	1003301.37
24/7/19	ASIA	Retail	C1	New	127342	3151780.41
10/7/19	CANADA	Shopify	F3	New	51	8844.93
26/6/19	OCEANIA	Retail	C3	New	152921	5551385.36
29/5/19	SOUTH AMERICA	Shopify	null	New	53	10056.2
22/8/18	AFRICA	Retail	null	Existing	31721	1718863.58
25/7/18	SOUTH AMERICA	Retail	null	New	2136	8

Segment	Age_band
1	Young Adults
2	Middle Aged
3 or 4	Retirees

Segment	Demographic
C	Couples
F	Families

Case Study 2

Solution:

{Concept used: Case statements}

```
Create table table1 as
Select *, case when right(segment,1) = 1 then 'Young Adults'
              When right(segment,1) = 2 then 'Middle Aged'
Case When right(segment,1) in (3,4) then 'Retirees' else null end as Age_Band,
case when left(segment,1) = 'C' then 'Couples'
      When left(segment,1) = 'F' then 'Families'
Else null end as Demographic

from clean_weekly_sales
```



Case Study 2

Question 3. As a part of further data exploration,What day of the week is used for each week_date value?

Note: Here you can use (optional)newly generated table from question 1.

Week_date	Region	Platform	Segment	Customer_type	Transactions	Sales
9/9/20	OCEANIA	Shopify	C3	New	610	110033.89
29/7/20	AFRICA	Retail	C1	New	110692	3053771.19
22/7/20	EUROPE	Shopify	C4	Existing	24	8101.54
13/5/20	AFRICA	Shopify	null	Guest	5287	1003301.37
24/7/19	ASIA	Retail	C1	New	127342	3151780.41
10/7/19	CANADA	Shopify	F3	New	51	8844.93
26/6/19	OCEANIA	Retail	C3	New	152921	5551385.36
29/5/19	SOUTH AMERICA	Shopify	null	New	53	10056.2
22/8/18	AFRICA	Retail	null	Existing	31721	1718863.58
25/7/18	SOUTH AMERICA	Retail	null	New	2136	8

Case Study 2

Solution:

{Concept used: date time functions}

```
Select *, day_of_week(week_date) as weekday from  
clean_weekly_sales
```



Case Study 2

Question 4. What is the total sales for each region for each month?

Note: Here you can use (optional) newly generated table from question 1.

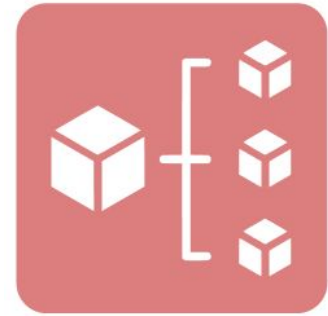
Week_date	Region	Platform	Segment	Customer_type	Transactions	Sales
9/9/20	OCEANIA	Shopify	C3	New	610	110033.89
29/7/20	AFRICA	Retail	C1	New	110692	3053771.19
22/7/20	EUROPE	Shopify	C4	Existing	24	8101.54
13/5/20	AFRICA	Shopify	null	Guest	5287	1003301.37
24/7/19	ASIA	Retail	C1	New	127342	3151780.41
10/7/19	CANADA	Shopify	F3	New	51	8844.93
26/6/19	OCEANIA	Retail	C3	New	152921	5551385.36
29/5/19	SOUTH AMERICA	Shopify	null	New	53	10056.2
22/8/18	AFRICA	Retail	null	Existing	31721	1718863.58
25/7/18	SOUTH AMERICA	Retail	null	New	2136	8

Case Study 2

Solution:

{Concept used: aggregate functions(sum)}

```
Select sum(sales),platform,month_number from  
Clean_weekly_sales  
Group by platform,month_number
```



Case Study 2

Question 5. Which age_band and demographic values contribute the most to Retail sales?

Note: Here you can use (optional) newly generated table from question 1.

Week_date	Region	Platform	Segment	Customer_type	Transactions	Sales
9/9/20	OCEANIA	Shopify	C3	New	610	110033.89
29/7/20	AFRICA	Retail	C1	New	110692	3053771.19
22/7/20	EUROPE	Shopify	C4	Existing	24	8101.54
13/5/20	AFRICA	Shopify	null	Guest	5287	1003301.37
24/7/19	ASIA	Retail	C1	New	127342	3151780.41
10/7/19	CANADA	Shopify	F3	New	51	8844.93
26/6/19	OCEANIA	Retail	C3	New	152921	5551385.36
29/5/19	SOUTH AMERICA	Shopify	null	New	53	10056.2
22/8/18	AFRICA	Retail	null	Existing	31721	1718863.58
25/7/18	SOUTH AMERICA	Retail	null	New	2136	8

Case Study 2

Solution:

{Concept used: subqueries, case statements, group by clause}

Select * from

```
(Select *, case when right(segment,1) = 1 then 'Young Adults'
               When right(segment,1) = 2 then 'Middle Aged'
Case When right(segment,1) in (3,4) then 'Retirees' else null end as Age_Band,
case when left(segment,1) = 'C' then 'Couples'
       When left(segment,1) = 'F' then 'Families'
Else null end as Demographic, sum(sales) as total_sales
```

from clean_weekly_sales

Where **platform= 'Retail'**

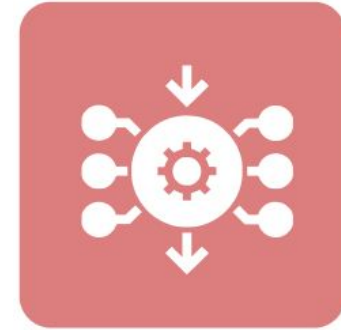
Group by

```
case when right(segment,1) = 1 then 'Young Adults'
       When right(segment,1) = 2 then 'Middle Aged'
When right(segment,1) in (3,4) then 'Retirees' else null end ,
Case When right(segment,1) in (3,4) then 'Retirees' else null end as Age_Band,
case when left(segment,1) = 'C' then 'Couples'
       When left(segment,1) = 'F' then 'Families'
Else null end
```

) base

Order by Total_sales

Limit 1



Case Study - 3



Case Study 3

Suppose you are working for Amazon and it is the data representing the support of Agents of that team.

Table Name- Snapshot: It gives the current status and details of all the cases ever created. When a case is reopened, then the ID of original case become the parentID of the reopened case. SolvedTimestamp is null for the cases which have not been solved yet.

Snapshot								
ID	ParentID	CreatedTimestamp	FirstResponseTime (mins)	SolvedTimestamp	AgentID	Rating	Vertical	Status
a	b	25-01-2021 12:00	45	null	XY		Profile	Customer response pending
c	b	01-13-2021 02:00:00	123	15-01-2021	AJ	2	Profile	Solved
b	-	01-01-2021 15:00	24	05-01-2021	YH	3	Profile	Solved
d	-	01-01-2021 15:00	230	14-05-2021	YW	4	UPI	Solved

Amazon has set up a centralized Customer Analytics Team who will be reporting to the CEO's office. The team has decided to impress the CEO's office by creating a dashboard with some important business/ operations metrics. Suppose you are a member of the analytics team then please help them define these metrics with the help of sql queries.

csv file- <https://drive.google.com/file/d/1gcx2JoD3eA016X8AbQAOafTea2pEd-5T/view?usp=sharing>

Case Study 3

Question 1. To track the performance or output of the team , it is important to check the inflow volume the team is receiving. The team has decided to calculate average cases received per day.Find this trend week over week

Snapshot								
ID	ParentID	CreatedTimestamp	FirstResponseTime (mins)	SolvedTimestamp	AgentID	Rating	Vertical	Status
a	b	25-01-2021 12:00	45	null	XY		Profile	Customer response pending
c	b	01-13-2021 02:00:00	123	15-01-2021	AJ	2	Profile	Solved
b	-	01-01-2021 15:00	24	05-01-2021	YH	3	Profile	Solved
d	-	01-01-2021 15:00	230	14-05-2021	YW	4	UPI	Solved

Case Study 3

Solution:

{Concepts used: Week and Count function}

```
Select  
week(createdtimestamp) as week_number,  
count(distinct(id))/7  
From snapshot  
Group by week(solvedtimestamp)
```



Case Study 3

Question 2 .First response time is an important metric to comfort the customer. Suppose you order something from an ecommerce platform and reach out to them with a refund related request. You don't hear back from them for a few hours/days. It will lead to a bad customer experience and loss of trust in the platform.Keeping this in mind, Amazon has decided the ideal first response time is ≤ 120 mins.Calculate the % of cases which were reached out by the agent within the ideal time for each week

Snapshot								
ID	ParentID	CreatedTimestamp	FirstResponseTime (mins)	SolvedTimestamp	AgentID	Rating	Vertical	Status
a	b	25-01-2021 12:00	45	null	XY		Profile	Customer response pending
c	b	01-13-2021 02:00:00	123	15-01-2021	AJ	2	Profile	Solved
b	-	01-01-2021 15:00	24	05-01-2021	YH	3	Profile	Solved
d	-	01-01-2021 15:00	230	14-05-2021	YW	4	UPI	Solved

Case Study 3

Solution:

{Concepts used: week,distinct and count function and case statements}

```
Select
week(createdtimestamp) as week_number,
sum(case when FirstResponseTime(mins) <= 120 then 1 else 0 end) as FRT_cases)/count(distinct(id)) as FRT%
From snapshot
Group by week(createdtimestamp)
```



Case Study 3

Question 3. Overall Handling Time is also an important metric to track agent/business performance. This metric is also used to avoid churn. Suppose the ideal case closed time is ≤ 2 days. Calculate the % of cases which were resolved with the ideal time

Snapshot								
ID	ParentID	CreatedTimestamp	FirstResponseTime (mins)	SolvedTimestamp	AgentID	Rating	Vertical	Status
a	b	25-01-2021 12:00	45	null	XY		Profile	Customer response pending
c	b	01-13-2021 02:00:00	123	15-01-2021	AJ	2	Profile	Solved
b	-	01-01-2021 15:00	24	05-01-2021	YH	3	Profile	Solved
d	-	01-01-2021 15:00	230	14-05-2021	YW	4	UPI	Solved

Case Study 3

Solution:

{concepts used: week function, subtracting 2 dates and case statements}

```
Select
week(createdtimestamp) as week_number,
sum(case when datediff(day,solvedtimestamp,createdtimestamp) <= 120 then 1 else 0 end) as ART_cases)/count(distinct(id))
as ART%
From snapshot
Group by week(createdtimestamp)
```



Case Study 3

Question 4. The customer dissatisfaction can be reflected by the number of times the customer reopens the case. Suppose you raise a request on an ecommerce platform and the request is closed by the agent. If you reopen the case or reply on the same message thread the case is considered to be reopened.Help the team write a query for the reopen% of the cases i.e cases which were not first contact resolution.Find this metric week over week so that dashboard can have all the metrics at the same granularity.

Snapshot								
ID	ParentID	CreatedTimestamp	FirstResponseTime (mins)	SolvedTimestamp	AgentID	Rating	Vertical	Status
a	b	25-01-2021 12:00	45	null	XY		Profile	Customer response pending
c	b	01-13-2021 02:00:00	123	15-01-2021	AJ	2	Profile	Solved
b	-	01-01-2021 15:00	24	05-01-2021	YH	3	Profile	Solved
d	-	01-01-2021 15:00	230	14-05-2021	YW	4	UPI	Solved

Case Study 3

Solution:

{Concepts used: week,distinct and count function , case statements and left join}

```
Select  
week(createdtimestamp) as week_number,  
count(distinct(case when B.Parentid is not null then A.ID else null end) as reopen) / count(distinct(id)) as Reopen%  
  
from snapshot A  
left join snapshot B  
On A.Id = B.Parentid
```

Case Study 3

Question 5 . We also want to see if there are agents from a particular vertical who are not performing good.We want to see this metric on overall level till date.Calculate Average Rating for each Vertical on an overall level.

Snapshot								
ID	ParentID	CreatedTimestamp	FirstResponseTime (mins)	SolvedTimestamp	AgentID	Rating	Vertical	Status
a	b	25-01-2021 12:00	45	null	XY		Profile	Customer response pending
c	b	01-13-2021 02:00:00	123	15-01-2021	AJ	2	Profile	Solved
b	-	01-01-2021 15:00	24	05-01-2021	YH	3	Profile	Solved
d	-	01-01-2021 15:00	230	14-05-2021	YW	4	UPI	Solved

Case Study 3

Solution:

{concepts used: avg functions}

.

Select avg(Rating),Vertical from snapshot
Group by Vertical



Case Study - 4



Case Study 4

There is a data maintained for height of men and women of a country. The height of male and female is mentioned in cm as well as feet. R_id is a unique identifier assigned to a person. If the person is male then the female column is left blank and vice versa.

This data needs to be used to understand the physique of the population of the country

Table description – Height of Male and Female

In a **Height** table given below, we have R_id (int), Country_name (varchar), Male_cm (Male height in cm) (decimal), Fem_cm (Female height in cm) (decimal), Male_Ft (Male height in feet) (decimal) and Fem_Ft (Female height in feet) (decimal)

R_ID	Country_Name	Male_cm	Fem_cm	Male_ft	Fem_ft
1234	India	170.18		5'7	
1235	China		165.1		5'5
1236	India		177.8		5'10

csv file- <https://drive.google.com/file/d/1qGXL4DU4T10Ow9y9f6GvbdgawEm4bO1/view?usp=sharing>

Case Study 4

Question 1. It is becoming difficult to process the data. Also, it has been decided that the analysis will be done by dividing the total data into two equal parts. For this, write a query to fetch first 50% of the records from the "Height" table sorted by country name?

R_ID	Country_Name	Male_cm	Fem_cm	Male_ft	Fem_ft
1234	India	170.18		5'7	
1235	China		165.1		5'5
1236	India		177.8		5'10

Case Study 4

Solution:

{concepts used: top function}

```
SELECT  
TOP 50 PERCENT *  
FROM Height  
ORDER BY Country_name asc
```



Case Study 4

Question 2. The analysis is further refined to take random sub groups. Write a query to find the first three records from the "Height" table, where the country is starting from "Ge" and order it on the basis of height in cm?

R_ID	Country_Name	Male_cm	Fem_cm	Male_ft	Fem_ft
1234	India	170.18		5'7	
1235	China		165.1		5'5
1236	India		177.8		5'10

Case Study 4

Solution:

{concepts used: group by statements}

```
SELECT *  
FROM Height  
WHERE  
Country_name LIKE 'Ge%'  
ORDER BY Male_cm,Fem_cm LIMIT 3
```



Case Study 4

Question 3. To clean the data ,Write a query to create a new column called Male_cm_new in which null values are replaced by the average height of male in that country

R_ID	Country_Name	Male_cm	Fem_cm	Male_ft	Fem_ft
1234	India	170.18		5'7	
1235	China		165.1		5'5
1236	India		177.8		5'10

Case Study 4

Solution:

{concepts used: case statements, avg function and group by statement}

```
Select case when Male_cm is null then Average_Height_Men  
Else Male_cm else as Male_cm_new  
From (  
Select avg(Male_cm) as Average_Height_Men, country  
from Height  
group by country) base
```



Case Study 4

Question 4. identify the outliers in data. The outlier in the data are identified as the Male_cm or Female_cm having a difference from the overall average Male_cm and Female_cm by 100 cm or more

R_ID	Country_Name	Male_cm	Fem_cm	Male_ft	Fem_ft
1234	India	170.18		5'7	
1235	China		165.1		5'5
1236	India		177.8		5'10

Case Study 4

Solution:

{concepts used: case statements and subqueries}

```
Select case when abs(male_cm - average_male_height) >= 100 then 1
When abs(fem_cm - average_female_height) >= 100 then 1
Else 0 end as flag
(
    from Height
Select avg(male_cm) as average_male_height,
avg(female_cm) as average_female_height
from height) base
```



Case Study 4

Question 5. Write a query to find the height of most number of men to see the general trend

R_ID	Country_Name	Male_cm	Fem_cm	Male_ft	Fem_ft
1234	India	170.18		5'7	
1235	China		165.1		5'5
1236	India		177.8		5'10

Case Study 4

Solution:

{concepts used: group by statements and subqueries}

```
Select * from  
(Select count(*) as cnt , male_cm  
* from  
Height  
Group by male_cm) base  
Order by cnt  
Limit 1
```



Case Study - 5



Case Study 5

Please find the below table named Stud. The school wants to focus on health of the students and considers weight as an important metric to understand students performance in sports and their physical and mental well being

Table Name – Stud

StudID	Class	Class
A	8	50
B	9	60
A	8	55
B	9	70

csv file - https://drive.google.com/file/d/1kebDY6ngjsO_LwHpPm9Hg72AJHuXl9iZ/view?usp=sharing

Case Study 5

Question 1. Create a table name 'student' with two columns Student_id and weight. Create a table named Student with a column The Student_id column will be 'Stud ID' – 'Class'. This column studentid will be a unique identifier of the student

StudID	Class	Class
A	8	50
B	9	60
A	8	55
B	9	70

Case Study 5

Solution:

{concepts used: concat string function}

```
Create table Student as  
Select  
*,  
Concat(StudID , ' - ', Class)  
from stud
```



Case Study 5

Question 2. The school wants to identify the students with highest weight to consider them sample for their observations. So that there are no biases in the experiment the students are picked across all classes. Write a query to find the minimum number of such students such that some of their weights is ≥ 200

StudID	Class	Class
A	8	50
B	9	60
A	8	55
B	9	70

Case Study 5

Solution:

{concepts used: sum function,average function(aggregate/window)}

```
Select  
  * from (  
Select  
  *,  
  sum(weight) over (order by weight desc) as running sum  
  from student  
  ) base  
Where running_sum >= 200
```



Case Study 5

Question 3. In results for question 2 , it was observed that most of the students were from higher classes who weighed higher . So, the experiment was extended to incorporate sample from all classes. Write a query to find the minimum number of such students for each class such that some of their weights is ≥ 200

StudID	Class	Class
A	8	50
B	9	60
A	8	55
B	9	70

Case Study 5

Solution:

{concepts used: sum function,average function(aggregate/window)}

```
Select  
* from (  
Select  
*,  
sum(weight) over (partition by class order by weight desc) as running sum  
from stud  
) base  
Where running_sum >= 200
```



Case Study 5

Question 4. The school also wants to understand top how many students from each class are required to form a sample.They want to associate the students with rank.Write a query to rank the student in the descending order of their weights for each class

StudID	Class	Class
A	8	50
B	9	60
A	8	55
B	9	70

Case Study 5

Solution:

{concepts used: rank function and partition by(Windows function)}

Select

*,

rank() over (partition by class order by weight desc) as rnk
from stud



Case Study 5

Question 5. The school wants to divide the students into different groups on the basis of their class. They will then use this data to compare the weight of the particular group with such data for different schools. This will also be used to compare a student's weight with the average weight of the group he is/isn't part of. Write a query to find the average weight for each group where class 1,2 is **Group1**, Class 3,4 is a **Group2**, Class 5,6 is **Group3**, Class 7,8 is **Group4**, Class 9,10 is **Group5**. Show these group averages as separate columns

StudID	Class	Class
A	8	50
B	9	60
A	8	55
B	9	70

Case Study 5

Solution:

{Concepts used: case statements, numeric function(window)}

```
Select
Avg((Case when class in (1,2) then weight else null end) ) as Group1,
Avg((Case when class in (3,4) then weight else null end) ) as Group2,
Avg((Case when class in (5,6) then weight else null end) ) as Group3,
Avg((Case when class in (7,8) then weight else null end) ) as Group4,
Avg((Case when class in (9,10) then weight else null end) ) as Group5
from
stud
```



THANK YOU