Question 1:

Let’s say you have two tables, Businesses and Stores.

The Businesses table has 100,000+ rows. Here are the first 6 rows:

business_name	store_id
business_1	store_1
business_1	store_2
business_1	store_3
business_2	store_4
business_3	store_5
business_3	store_6
...	...

The Stores table has 100,000+ rows. Here are the first 6 rows:

store_id	total_deliveries
store_1	500
store_2	3000
store_3	25
store_4	62
store_5	8300
store_6	650
...	...

Create a SQL query that shows the top 10 businesses with the most deliveries.

Query:

```
SELECT B.business_name, SUM(S.total_deliveries) as sum_deliveries
FROM BUSINESS B
INNER JOIN STORES S
ON B.store_id = S.store_id
GROUP BY B.business_name
ORDER BY sum_deliveries DESC
LIMIT 10;
```

Question 2:

Let’s say you have two tables, Constellations, and Stars.

The Constellations table has 1,000+ rows. Here are the first 6 rows.

Constellation	Star
Scorpius	Antares A
Scorpius	λ Sco
Draco	γ Dra
Draco	η Dra
Draco	β Dra
Hydra	Alphard
...	...

The Stars table has 10,000+ rows. Here are the first 6 rows.

Star	Apparent_Magnitude
Antares A	0.91
λ Sco	1.62
γ Dra	2.24
η Dra	2.73
β Dra	2.79
Alphard	1.99
...	...

Find all constellations with an average apparent magnitude of 3 to 10.

Query:

```
SELECT C.Constellation, AVG(S.Apparent_Magnitude) as avg_apparent_magnitude
FROM Constellations C
INNER JOIN Stars S
ON C.Star = S.Star
GROUP BY C.Constellation
HAVING avg_apparent_magnitude >= 3 AND avg_apparent_magnitude <= 10;
```

Question 3:

Let’s say you have two tables Audit_Log and Event_Types.

The Audit_Log table has 1M+ rows. Here are the first 6 rows:

Employee_ID	Date	Event_Type_ID
18746352819	01-01-2020	5
38271948271	03-15-2020	8
37561739201	02-19-2020	5
27346183746	01-23-2020	150
52389174321	02-15-2020	100
72614321748	01-10-2020	5
...	...	...

The Event_Types table has 200 rows. Here are the first 6 rows:

Event_Type_ID	Event_Type
1	Create
2	Update
3	Delete
4	View
5	Share
6	Activate
...	...

The Audit_Log table has a row for each action that an employee takes.
Create a query to show which employees have the “Create” event more than 4000 times, but less than 5000 times.

Query:

```
SELECT Employee_ID
FROM Audit_Log
GROUP BY Employee_ID
HAVING Event_Type_ID =
(
SELECT Event_Type_ID
FROM Event_Types
WHERE Event_Type = "Create"
)
AND (COUNT(Event_Type_ID) >= 4000  AND COUNT(Event_Type_ID) <= 5000);
```

Question 4:

For the results in question 3, please explain how you would validate the results of your query.

Here are some of the Validation ideas:

1. I will write a query in a different way and compare the results (maybe using a join)
2. Another way is to run the query for other Event_Types and compare the numbers
3. I will bucket by dates and see if there are gaps (maybe by month, this is a crude way to tell if there’s missing data)
4. Combine (2) and (3), for example, plot monthly time series for Create Vs Delete and eyeball any anomalies.