Section – A: Theory Questions

1. **What is aggregation and how can it help for large databases?**
   Aggregation is the process of grouping data from multiple rows into a summarized form to better understand the dataset
   Key Aggregation Functions:
       SUM: returns the sum of values in a column.
       AVG: returns the average of values in a column.
       MIN: returns the minimum value in a column.
       MAX: returns the maximum value in a column.
       COUNT: returns the number of values in a column.
   By aggregating data, you can obtain a clearer picture of the underlying patterns and trends in large datasets.
   These functions allow you to summarize and analyze data in a meaningful way, making it easier to understand and draw insights from large datasets

2. **How do you find the count of distinct values for a column and the count of columns containing null values in SQL?**
   SELECT COUNT(DISTINCT column) as "Distinct Count", COUNT(column) as "Total Count", COUNT(column) - COUNT(DISTINCT column) as "Null Count" FROM table;

3. **What is the difference between "COUNT(*)" and "COUNT(column)" in SQL?**
   "COUNT()" returns the total number of rows in the result set, regardless of whether or not they contain null values,
   whereas "COUNT(column)" only returns the number of rows where the specified column contains a non-null value.

4. **What is the flow of the execution of aggregation?**
   from ⟶ where ⟶ groupby ⟶ having ⟶ select ⟶ order by
   tables ⟶ filtering columns ⟶ groupping columns ⟶ filtering grouping columns ⟶
   show results ⟶ sort the final results based on columns

5. Draw a pictorial representation of grouped customers who placed the orders

For example, the data present in the below table named "Orders"

order_id	order_name	price	customer_id
1	"Pizza"	3.34	5
2	"Burger"	12.5	1
3	"Noodles"	6.6	1
4	"Hot-Dog"	2.28	3
5	"Fried Rice"	5.12	3
6	"Chicken 65"	15.2	4
7	"Sch.Soup"	1.04	4
8	"Manchuria"	5.28	3
9	"Dumplings"	4.52	2

1. filter each customer based on customer_id and show order_name and price
2. find min, max, count, avg, the sum of prices in the orders table (group customers based on customer_id)

individual customers ⟶ filter the specific customer by where clause
select order_name, price from orders where customer_id = 1;

order_name	price
"Burger"	12.5
"Noodles"	6.6

select order_name, price from orders where customer_id = 2;

order_name	price
"Dumplings"	4.52

select order_name, price from orders where customer_id = 3;

order_name	price
"Hot-Dog"	2.28
"Fried Rice"	5.12
"Manchuria"	5.28

select order_name, price from orders where customer_id = 4;

order_name	price
"Chicken 65"	15.2
"Sch.Soup"	1.04

select order_name, price from orders where customer_id = 5;

order_name	price
"Pizza"	3.34

groupby -> customer_id
Find 5-point summary (min, max, sum, avg, count)
select customer_id, count(*), min(price), max(price), sum(price), avg(price) from orders
groupby customer_id;

customer_id	count(*)	min(price)	max(price)	sum(price)	avg(price)
1	2	6.6	12.5	19.1	9.55
2	1	4.52	4.52	4.52	4.52
3	3	2.28	5.28	12.68	4.23
4	2	1.04	15.2	16.24	8.12
5	1	3.34	3.34	3.34	3.34

6. **What is the average formula write an SQL query to find the average of column "price"**

Average Formula:

sum of observations / no of observations

Average Query:

select (sum(column) / count(column)) as average from table_name

Example:

select avg(price) from orders

7. **What is the purpose of having instead of where clause in SQL?**

The "WHERE" clause filters individual records based on specified conditions, whereas the "HAVING" clause filters groups of records created by the "GROUP BY" clause, based on aggregate values.

Syntax:

select sum(column2) from table_name where column1 < value;
select sum(column2) from table_name groupby column1 having sum(column2) < value;
select sum(column2) from table_name having sum(column) < value;
select sum(column2) from table_name where column2 groupby column1 having sum(column2) < value;