Section 1

The company wants to analyze its target audience. Thus the company needs to analyze the purchasing behavior of customers in the region along with high-ticket customers to ensure retention.

1. Find the names of all customers who made an order on their first visit to the website.

```
SELECT
c.name AS customer_name
FROM
customers c
JOIN
orders_0 o ON c.customer_id = o.customer_id
WHERE
o.order_date = (SELECT MIN(order_date) FROM orders_0 WHERE customer_id = c.customer_id);
```

2. Find the names of customers who spent more than the average total amount spent by all customers.

```
SELECT c.name
FROM customers c
JOIN orders_0 o ON c.customer_id = o.customer_id
WHERE o.total_amount > (SELECT AVG(total_amount) FROM orders_0);
```

3.Find the names and cities of customers who have made an order with a total amount greater than the average total amount spent by customers in their city.

```
SELECT c.name, c.city
FROM customers c
JOIN orders_0 o ON c.customer_id = o.customer_id
WHERE o.total_amount > (
SELECT AVG(o2.total_amount)
FROM orders_0 o2
JOIN customers c2 ON o2.customer_id = c2.customer_id
WHERE c2.city = c.city
);
```

3.Find the names of all customers who have ordered a product with a price greater than ₹ 8000.

```
select c.name from customers c
where customer_id in
(select customer_id from orders_0
where order_id in
(select order_id from order_details
where product_id in
(select product_id from products
where price > 8000 )));
```

Section 2

The company wants to analyze its product-market fit and wants to explore its product in Terms of price, selling ability, and revenue generation.

1. Find the names of all customers who have ordered at least one product that no Other customer has ordered.

```
SELECT DISTINCT c1.name
FROM customers c1
JOIN orders_0 o1 ON c1.customer_id = o1.customer_id
JOIN order_details od1 ON o1.order_id = od1.order_id
WHERE NOT EXISTS (
SELECT 1
FROM customers c2
JOIN orders_0 o2 ON c2.customer_id = o2.customer_id
JOIN order_details od2 ON o2.order_id = od2.order_id
WHERE c2.customer_id != c1.customer_id
AND od2.product_id = od1.product_id
);
```

Section 3

The HR team is interested in analyzing employee profiles of the company to ensure smooth functioning across departments.

1. Find the names of all employees whose name starts with the letter "J".

```
SELECT name
FROM employees_new
WHERE name LIKE 'J%';
```

2. Find the names of all employees who were hired before January 1st, 2000, and Whose salary is greater than ₹ 5,00,000?

```
SELECT name
FROM employees_new
WHERE hire_date < '2000-01-01' AND salary > 500000;
```

Section 4

The sales team is interested in understanding the Product Market Fit and also analyzing the price sensitivity of the market. Help them with the following queries -

1. Find the names of all customers who have ordered both "Product A" and "Product B".

```
SELECT c.name
FROM customers c
WHERE EXISTS (
SELECT 1
FROM orders_0 o
JOIN order_details od ON o.order_id = od.order_id
JOIN products p ON od.product_id = p.product_id
WHERE c.customer_id = o.customer_id
AND p.name IN ('Product A', 'Product B')
GROUP BY o.order_id
HAVING COUNT(DISTINCT p.name) = 2
);
```

2. Find the names of all customers who have not ordered any product with a price greater than ₹ 8000.

```
SELECT c.name
FROM customers c
WHERE NOT EXISTS (
SELECT 1
FROM orders_0 o
JOIN order_details od ON o.order_id = od.order_id
JOIN products p ON od.product_id = p.product_id
WHERE c.customer_id = o.customer_id
AND p.price > 8000
);
```

3. Find the names of all customers who have ordered a total quantity of at least 10 units of "Product C"

```
SELECT c.name
FROM customers c
WHERE EXISTS (
SELECT 1
FROM orders_0 o
JOIN order_details od ON o.order_id = od.order_id
JOIN products p ON od.product_id = p.product_id
WHERE c.customer_id = o.customer_id
AND p.name = 'Product C'
GROUP BY c.customer_id
HAVING SUM(od.quantity) >= 10
);
```

Section 5

The company wants to analyze basic details about its customers.

1. Find the names of all cities where at least one customer lives, in alphabetical order.

```
SELECT DISTINCT city
FROM customers
ORDER BY city;
```

2. Find the names of the first 5 customers whose names start with the letter "J".

```
SELECT name
FROM customers
WHERE name LIKE 'J%'
LIMIT 5;
```

3. Find the distinct first letters of all customer names, in alphabetical order.

```
SELECT DISTINCT SUBSTRING(name, 1, 1) AS first_letter
FROM customers
ORDER BY first_letter;
```