Q1)How would you fetch details of the customers who cancelled orders?
Soln.

```
SELECT cutomers_name
FROM customers c
LEFT JOIN orders o ON c.cutomer_id = o.customer_id
WHERE o.status = "cancelled ";
```

Q2) Fetch the details of customers who have done payments between the amount 5,000 and 35,000?
Soln.

```
SELECT customer_name FROM customers c
LEFT JOIN payments p
ON c.customer_id = p.customer_id
WHERE amount BETWEEN 5000 AND 35000;
```

Q3)Add new employee/salesman with following details:-
EMP ID-15657
First Name : Lakshmi
Last Name: Roy
Extension : x4065
Email:lakshmiroy1@lcomotors.comOffice
Code: 4
Reports To: 1088
Job Title: Sales Rep

Soln.

```
INSERT INTO employees (
      employee_id,
      last_name,
      first_name,
```

```
       extension,
       email,
       office_code,
       reports_to,
       job_title
   )
VALUES (
       15657,
       'Lakshmi',
       'Roy',
       'x4065',
       'lakshmiroy1@lcomotors.comOffice',
       '4',
       1088,
       'Sales Rep'
   );
```

Q4)Assign the new employee to the customer whose phone is 2125557413 .
Soln.

```
UPDATE customers
SET sales_employee_id = 15657
WHERE phone = '2125557413';
```

Q5)Write a SQL query to fetch shipped motorcycles.

Soln.

```
SELECT product_name
FROM products p
   LEFT JOIN orderdetails o ON p.product_code = o.product_code
   LEFT JOIN orders os ON o.order_id = os.order_id
WHERE p.product_line = 'Motorcycles'
```

```
   AND os.status = 'shipped';
```

Q6)Write a SQL query to get details of all employees/salesmen in the office located in Sydney.
Soln.

```
SELECT CONCAT(first_name, ' ', last_name) as full_name
FROM employees e
   LEFT JOIN offices o ON e.office_code = o.office_code
WHERE city = 'Sydney';
```

Q7)How would you fetch the details of customers whose orders are in process?
Soln.

```
SELECT customer_name
FROM customers c
   LEFT JOIN orders o ON c.customer_id = o.customer_id
WHERE o.status = 'In Process';
```

Q8)How would you fetch the details of products with less than 30 orders?
Soln.

```
SELECT product_name
FROM products p
   LEFT JOIN orderdetails o ON p.product_code = o.product_code
WHERE o.quantity_ordered < 30;
```

Q9)It is noted that the payment (check number OM314933) was actually 2575. Update the record.
Soln.

```sql
UPDATE payments
SET amount = 2575
WHERE check_number = 'OM314933';
```

Q10)Fetch the details of salesmen/employees dealing with customers whose orders are resolved.
Soln.

```sql
SELECT employee_id,
   CONCAT(e.first_name, ' ', e.last_name) as full_name
FROM employees e
   LEFT JOIN customers c ON e.employee_id = c.sales_employee_id
   LEFT JOIN orders o ON c.customer_id = o.customer_id
WHERE o.status = 'Resolved';
```

Q11)Get the details of the customer who made the maximum payment.
Soln.

```sql
SELECT customer_name
FROM customers c
   LEFT JOIN payments p ON c.customer_id = p.customer_id
WHERE p.amount = (
      SELECT MAX(amount)
      FROM payments
   );
```

Q12)Fetch list of orders shipped to France.
Soln.

```sql
SELECT o.order_id,
   c.customer_name
FROM orders o
   LEFT JOIN customers c ON o.customer_id = c.customer_id
```

```
WHERE o.status = 'shipped'
   AND country = 'France';
```

Q13)How many customers are from Finland who placed orders.
Soln.

```
SELECT COUNT(c.customer_id) as total_count
FROM customers c
   LEFT JOIN orders o ON c.customer_id = o.customer_id
WHERE c.country = 'Finland';
```

Q14)Get the details of the customer who made the maximum payment.
Soln.

```
SELECT c.customer_name,
   p.amount
FROM customers c
   LEFT JOIN payments p ON c.customer_id = p.customer_id
ORDER BY p.amount DESC
LIMIT 10;
```

Q15)Get the details of the customer and payments they made between May 2019 and June2019.

```
SELECT c.customer_name,
   p.payment_date,
   p.amount
FROM customers c
   LEFT JOIN payments p ON c.customer_id = p.customer_id
WHERE p.payment_date BETWEEN '2019-05-01' AND '2019-06-30'
ORDER BY 2;
```

Q16)How many orders are shipped to Belgium in 2018?
Soln.

```
SELECT COUNT(o.order_id) AS total_count
FROM orders o
   LEFT JOIN customers c ON o.customer_id = c.customer_id
WHERE country = 'Belgium'
   AND YEAR(o.shipped_date) = '2018';
```

Q17)Get the details of the salesman/employee with offices dealing with customers in Germany.

```
SELECT DISTINCT(e.employee_id),
   CONCAT(e.first_name, ' ', e.last_name) as full_name,
   e.job_title
FROM employees e
   LEFT JOIN customers c ON e.employee_id = c.sales_employee_id
WHERE c.country = 'Germany';
```

Q18)The customer (id:496) made a new order today and the details are as follows:
Order id : 10426
Product Code: S12_3148
Quantity : 41
Each price : 151
Order line number : 11
Order date : <today's date>
Required date: <10 days from today>
Status: In Process
Soln.

```
INSERT INTO orderdetails (
       order_id,
       product_code,
       quantity_ordered,
       each_price,
       order_line_number
```

```
   )
VALUES (10426, 'S12_3148', 41, 151, 11);
INSERT INTO orders (
       order_id,
       order_date,
       required_date,
       status,
       customer_id
   )
VALUES (
       10426,
       CURRENT_DATE(),
       (CURRENT_DATE() + INTERVAL 10 DAY),
       'In Process',
       496
   );
```

Q19)**Fetch details of employees who were reported for the payments made by the customers between June 2018 and July 2018.**

Soln.

```
SELECT e.employee_id,
   CONCAT(e.first_name, ' ', e.last_name) as full_name
from employees e
   LEFT JOIN customers c ON e.employee_id = c.sales_employee_id
   LEFT JOIN payments p ON c.customer_id = p.customer_id
WHERE p.payment_date BETWEEN '2018-06-01' AND '2018-07-31';
```

Q20)A new payment was done by a customer(id: 119). Insert the below details.

Check Number : OM314944

Payment date : <today's date>

Amount : 33789.55

Soln.

```sql
INSERT INTO payments (customer_id, check_number, payment_date, amount)
VALUES (119, 'OM314944', CURRENT_DATE(), 33789.55);
```

Q21)Get the address of the office of the employees that reports to the employee whose id is 1102.
Soln.

```sql
SELECT o.address_line1,
   o.address_line2
FROM employees e
   LEFT JOIN offices o ON e.office_code = o.office_code
WHERE e.reports_to = 1102;
```

Q22)Get the details of the payments of classic cars.
Soln.

```sql
SELECT pa.customer_id,
   pa.check_number,
   pa.payment_date,
   pa.amount
   FROM payments pa
   LEFT JOIN orders o ON pa.customer_id = o.customer_id
   LEFT JOIN orderdetails od ON o.order_id = od.order_id
   LEFT JOIN products p ON od.product_code = p.product_code
WHERE p.product_line = 'Classic Cars';
```

Q23)How many customers ordered from the USA?
Soln.

```sql
SELECT COUNT(c.customer_id) AS total_count
FROM customers c
   LEFT JOIN orders o ON c.customer_id = o.customer_id
WHERE country = 'USA';
```

Q24)Get the comments regarding resolved orders.
Soln.

```sql
SELECT comments
FROM orders
WHERE status = 'Resolved';
```

Q25)Fetch the details of employees/salesmen in the USA with office addresses.
Soln.

```sql
SELECT e.employee_id,
   CONCAT(e.first_name, ' ', e.last_name) AS full_name,
   o.country
FROM employees e
   LEFT JOIN offices o ON e.office_code = o.office_code
WHERE o.country = 'USA';
```

Q26)Fetch total price of each order of motorcycles. (Hint: quantity x price for each record).
Soln.

```sql
SELECT p.product_name,
   (o.quantity_ordered * o.each_price) AS total_cost
FROM orderdetails o
   LEFT JOIN products p ON o.product_code = p.product_code
WHERE p.product_line = 'Motorcycles';
```

Q27)Get the total worth of all planes ordered.
Soln.

```
SELECT SUM(o.quantity_ordered * o.each_price) AS total_cost
FROM orderdetails o
   LEFT JOIN products p ON o.product_code = p.product_code
WHERE p.product_line = 'Planes'
GROUP BY p.product_line;
```

Q28)How many customers belong to France?
Soln.

```
SELECT COUNT(customer_id) AS total_count
FROM customers
WHERE country = 'FRANCE';
```

Q29)Get the payments of customers living in France.
Soln.

```
SELECT c.customer_id,
   p.check_number,
   p.amount,
   p.payment_date AS total_count
FROM customers c
   LEFT JOIN payments p ON c.customer_id = p.customer_id
WHERE c.country = 'FRANCE';
```

Q30)Get the office address of the employees/salesmen who report to employee 1143.

Soln.

```
SELECT o.address_line1,
   o.address_line2
FROM offices o
   LEFT JOIN employees e ON o.office_code = e.office_code
WHERE e.reports_to = 1143;
```