Basic format

Creating the table

```
CREATE TABLE reaction (
   id INT AUTO_INCREMENT PRIMARY KEY,
user_nameVARCHAR(100),
reaction_typeVARCHAR(20),
post_id INT,
created_at DATETIME,
   location VARCHAR(100),
mood_level INT,
   comment TEXT
);
```

Inserting the data

```
INSERT INTO reaction (user_name, reaction_type, post_id, created_at, location, mood_level, comment) VALUES
('Alice', 'like', 101, '2025-08-07 10:15:00', 'New York', 8, 'Nice post!'),
('Bob', 'love', 102, '2025-08-06 14:20:00', 'Los Angeles', 9, 'Awesome!'),
('Charlie', 'angry', 101, '2025-08-05 09:10:00', 'Chicago', 3, NULL),
('Diana', 'wow', 103, '2025-08-07 08:30:00', 'Miami', 7, 'Interesting point.'),
('Ethan', 'sad', 104, '2025-08-04 16:45:00', 'Dallas', 2, NULL),
('Fiona', 'love', 102, '2025-08-07 12:00:00', 'Boston', 6, 'Well said.'),
('George', 'like', 105, '2025-08-03 11:25:00', 'Seattle', 5, NULL),
('Hannah', 'like', 106, '2025-08-07 15:50:00', 'Denver', 9, 'Completely agree!'),
('Ian', 'angry', 107, '2025-08-06 13:15:00', 'Phoenix', 4, 'Not okay with this.'),
('Jane', 'sad', 108, '2025-08-02 18:40:00', 'Atlanta', 1, NULL),
('Sam', 'like', 101, '2025-08-07 10:50:00', 'New York', 8, NULL),
('Anita', 'wow', 109, '2025-08-01 20:10:00', 'Houston', 7, NULL),
('Brian', 'love', 110, '2025-08-07 09:05:00', 'San Francisco', 10, 'Fantastic work!'),
('Catherine', 'like', 111, '2025-08-07 08:55:00', 'New York', 6, NULL),
```

('Daniel', 'angry', 112, '2025-08-05 17:35:00', 'Chicago', 2, NULL);

mysql> select * from reaction;

```
+----+-----------+---------------+---------+---------------------+--------------+------------+---------------------+
| id | user_name | reaction_type | post_id | created_at          | location     | mood_level | comment             |
+----+-----------+---------------+---------+---------------------+--------------+------------+---------------------+
|  1 | Alice     | like          |     101 | 2025-08-07 10:15:00 | New York     |          8 | Nice post!          |
|  2 | Bob       | love          |     102 | 2025-08-06 14:20:00 | Los Angeles  |          9 | Awesome!            |
|  3 | Charlie   | angry         |     101 | 2025-08-05 09:10:00 | Chicago      |          3 | NULL                |
|  4 | Diana     | wow           |     103 | 2025-08-07 08:30:00 | Miami        |          7 | Interesting point.  |
|  5 | Ethan     | sad           |     104 | 2025-08-04 16:45:00 | Dallas       |          2 | NULL                |
|  6 | Fiona     | love          |     102 | 2025-08-07 12:00:00 | Boston       |          6 | Well said.          |
|  7 | George    | like          |     105 | 2025-08-03 11:25:00 | Seattle      |          5 | NULL                |
|  8 | Hannah    | like          |     106 | 2025-08-07 15:50:00 | Denver       |          9 | Completely agree!   |
|  9 | Ian       | angry         |     107 | 2025-08-06 13:15:00 | Phoenix      |          4 | Not okay with this. |
| 10 | Jane      | sad           |     108 | 2025-08-02 18:40:00 | Atlanta      |          1 | NULL                |
| 11 | Sam       | like          |     101 | 2025-08-07 10:50:00 | New York     |          8 | NULL                |
| 12 | Anita     | wow           |     109 | 2025-08-01 20:10:00 | Houston      |          7 | NULL                |
| 13 | Brian     | love          |     110 | 2025-08-07 09:05:00 | San Francisco |         10 | Fantastic work!     |
| 14 | Catherine | like          |     111 | 2025-08-07 08:55:00 | New York     |          6 | NULL                |
| 15 | Daniel    | angry         |     112 | 2025-08-05 17:35:00 | Chicago      |          2 | NULL                |
+----+-----------+---------------+---------+---------------------+--------------+------------+---------------------+
```

LIKE

Write a query to find all reactions made by users whose names start with the letter 'A'.

mysql> select * from reaction where user_name like 'a%';

```
+----+-----------+---------------+---------+---------------------+----------+------------+------------+
| id | user_name | reaction_type | post_id | created_at          | location | mood_level | comment    |
+----+-----------+---------------+---------+---------------------+----------+------------+------------+
|  1 | Alice     | like          |     101 | 2025-08-07 10:15:00 | New York |          8 | Nice post! |
| 12 | Anita     | wow           |     109 | 2025-08-01 20:10:00 | Houston  |          7 | NULL       |
+----+-----------+---------------+---------+---------------------+----------+------------+------------+
```

AS (Alias)

Write a query to display user_name as Reactor and reaction_type as Type.

mysql> select user_name as Reactor, reaction_type as Type from reaction;

```
+-----------+-------+
| Reactor   | Type  |
+-----------+-------+
| Alice     | like  |
| Bob       | love  |
| Charlie   | angry |
| Diana     | wow   |
| Ethan     | sad   |
| Fiona     | love  |
| George    | like  |
| Hannah    | like  |
| Ian       | angry |
| Jane      | sad   |
| Sam       | like  |
| Anita     | wow   |
| Brian     | love  |
| Catherine | like  |
| Daniel    | angry |
+-----------+-------+
```

BETWEEN

Write a query to find all reactions where the mood level is between 4 and 8.

```
mysql>    select * from reaction where  mood_level between 4 and 8;
+----+-----------+---------------+---------+---------------------+----------+------------+---------------------+
| id | user_name | reaction_type | post_id | created_at          | location | mood_level | comment             |
+----+-----------+---------------+---------+---------------------+----------+------------+---------------------+
|  1 | Alice     | like          |     101 | 2025-08-07 10:15:00 | New York |          8 | Nice post!          |
|  4 | Diana     | wow           |     103 | 2025-08-07 08:30:00 | Miami    |          7 | Interesting point.  |
|  6 | Fiona     | love          |     102 | 2025-08-07 12:00:00 | Boston   |          6 | Well said.          |
|  7 | George    | like          |     105 | 2025-08-03 11:25:00 | Seattle  |          5 | NULL                |
|  9 | Ian       | angry         |     107 | 2025-08-06 13:15:00 | Phoenix  |          4 | Not okay with this. |
| 11 | Sam       | like          |     101 | 2025-08-07 10:50:00 | New York |          8 | NULL                |
| 12 | Anita     | wow           |     109 | 2025-08-01 20:10:00 | Houston  |          7 | NULL                |
| 14 | Catherine | like          |     111 | 2025-08-07 08:55:00 | New York |          6 | NULL                |
+----+-----------+---------------+---------+---------------------+----------+------------+---------------------+
```

AND

Write a query to get all reactions where the type is 'like' and the mood level is above 7.

```
mysql> select reaction_type from reaction where reaction_type = 'like' and mood_level>7;
+---------------+
| reaction_type |
+---------------+
| like          |
| like          |
| like          |
+---------------+
```

OR

Write a query to list all reactions where the type is either 'love' or 'angry'.

```
mysql> select reaction_type from reaction where reaction_type='love' or 'angry';
+---------------+
| reaction_type |
+---------------+
```

| love |

| love |

| love |

+---------------+

NOT

Write a query to show all reactions that are not of type 'sad'.

mysql> select reaction_type from reaction where not reaction_type='sad';

+---------------+

| reaction_type |

+---------------+

| like |

| love |

| angry |

| wow |

| love |

| like |

| like |

| angry |

| like |

| wow |

| love |

| like |

| angry |

+---------------+

IN

Write a query to find all reactions where the reaction type is 'like', 'wow', or 'love'.

mysql> select reaction_type from reaction where reaction_type in ('like','wow','love');

+---------------+

| reaction_type |

+---------------+

| like |

| love |

| wow |

| love |

| like |

| like |

| like |

| wow |

| love |

| like |

+---------------+

IS NULL

Write a query to display all reactions where the comment is missing (i.e., NULL).

mysql> select comment from reaction where comment is null;

+---------+

| comment |

+---------+

| NULL |

| NULL |

| NULL |

| NULL |

| NULL |

| NULL |

| NULL |

| NULL |

+---------+

IS NOT NULL

Wrmysql> select comment from reaction where comment is not null;

+---------------------+

| comment |

+---------------------+

| Nice post! |

| Awesome! |

| Interesting point. |

| Well said. |

| Completely agree! |

| Not okay with this. |

| Fantastic work! |

+----------+.

UPPER()

Write a query to show all usernames in uppercase.

mysql> select upper(user_name) from reaction;

+----------+

| upper(user_name) |

+----------+

| ALICE |

| BOB |

| CHARLIE |

| DIANA |

| ETHAN |

| FIONA |

| GEORGE |

| HANNAH |

| IAN |

| JANE |

| SAM |

| ANITA |

| BRIAN |

| CATHERINE |

| DANIEL |

+----------+

LOWER()

Write a query to show all reaction types in lowercase.

mysql> select lower(reaction_type) from reaction;

+----------+

| lower(reaction_type) |

+----------------------+

| like |

| love |

| angry |

| wow |

| sad |

| love |

| like |

| like |

| angry |

| sad |

| like |

| wow |

| love |

| like |

| angry |

+----------------------+

LENGTH()

Wmysql> select length(user_name),length(user_name) as len from reaction where length(user_name)>6;

+-------------------+------+

| length(user_name) | len |

+-------------------+------+

| 7 | 7 |

| 9 | 9 |

+-------------------+------+rite a query to find all users whose usernames are longer than 6 characters.

NOW()ite a query to find all reactions made today (based on the created_at column).

Wrmysql> select now() from reaction;

```
+---------------------+
| now()               |
+---------------------+
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
| 2025-08-07 11:41:14 |
+---------------------+
```

DATE()

Write a query to extract and show only the date part from the created_at column.

```
mysql> select curdate() from reaction;
+------------+
| curdate()  |
+------------+
| 2025-08-07 |
| 2025-08-07 |
| 2025-08-07 |
| 2025-08-07 |
| 2025-08-07 |
```

| 2025-08-07 |

| 2025-08-07 |

| 2025-08-07 |

| 2025-08-07 |

| 2025-08-07 |

| 2025-08-07 |

| 2025-08-07 |

| 2025-08-07 |

| 2025-08-07 |

| 2025-08-07 |

+------------+

LIKE with %
Write a query to find all users whose name contains the substring 'an'.

mysql> select user_name from reaction where user_name like '%an%';

+-----------+

| user_name |

+-----------+

| Diana |

| Ethan |

| Hannah |

| Ian |

| Jane |

| Anita |

| Brian |

| Daniel |

+-----------+

ROUND()
Write a query to display each user's mood level rounded to the nearest 5.

mysql> select round(mood_level,5) from reaction;

+---------------------+

| round(mood_level,5) |

+---------------------+

| 8 |

| 9 |

| 3 |

| 7 |

| 2 |

| 6 |

| 5 |

| 9 |

| 4 |

| 1 |

| 8 |

| 7 |

| 10 |

| 6 |

| 2 |

+----------+

SUBSTR + UPPER()

Write a query to show the first 2 letters of each user's name in uppercase.

mysql> select substr(upper(user_name),1,2) from reaction;

+------------------------------+

| substr(upper(user_name),1,2) |

+------------------------------+

| AL |

| BO |

| CH |

| DI |

| ET |

| FI |

| GE |

| HA |

| IA |

| JA |

| SA |

| AN |

| BR |

| CA |

| DA |

+------------------------------+

NOT IN (with values)
Write a query to find all reactions not made on posts with IDs 10, 20, or 30.

mysql> select post_id from reaction where post_id not in ('10','20','30');

+---------+

| post_id |

+---------+

| 101 |

| 102 |

| 101 |

| 103 |

| 104 |

| 102 |

| 105 |

| 106 |

| 107 |

| 108 |

| 101 |

| 109 |

| 110 |

| 111 |

| 112 |

+---------+

COUNT and IS NULL
Write a query to count how many reactions have no comment.

mysql> select count(*) as no_comment_count from reaction where comment is null;

```
+------------------+
| no_comment_count |
+------------------+
|                8 |
+------------------+
```

Create the order table

```
CREATE TABLE orders (
order_id INT AUTO_INCREMENT PRIMARY KEY,
customer_nameVARCHAR(100),
product_nameVARCHAR(100),
order_date DATETIME,
   quantity INT,
   price DECIMAL(10,2),
   status VARCHAR(20),
shipping_address TEXT
);
```

Insert the data

```
INSERT INTO orders (customer_name, product_name, order_date, quantity, price, status, shipping_address) VALUES
('Alice', 'Laptop', '2025-08-07 09:00:00', 1, 1200.00, 'shipped', '123 Main St, New York'),
('Bob', 'Phone', '2025-08-06 14:10:00', 2, 650.50, 'pending', '456 Elm St, LA'),
('Charlie', 'Tablet', '2025-08-05 11:25:00', 1, 300.00, 'cancelled', NULL),
('Diana', 'Monitor', '2025-08-07 15:40:00', 3, 199.99, 'shipped', '789 Pine St, Chicago'),
('Ethan', 'Keyboard', '2025-08-03 08:20:00', 5, 49.99, 'processing', NULL),
('Fiona', 'Mouse', '2025-08-04 10:30:00', 4, 25.00, 'shipped', '321 Oak St, Houston');
```

```
mysql> select * from orders;
```

```
+----------+---------------+-------------+--------------------+----------+---------+------------+----------------------+
| order_id | customer_name | product_name | order_date         | quantity | price   | status     | shipping_address     |
+----------+---------------+-------------+--------------------+----------+---------+------------+----------------------+
|        1 | Alice         | Laptop      | 2025-08-07 09:00:00 |        1 | 1200.00 | shipped    | 123 Main St, New York |
|        2 | Bob           | Phone       | 2025-08-06 14:10:00 |        2 |  650.50 | pending    | 456 Elm St, LA       |
|        3 | Charlie       | Tablet      | 2025-08-05 11:25:00 |        1 |  300.00 | cancelled  | NULL                 |
|        4 | Diana         | Monitor     | 2025-08-07 15:40:00 |        3 |  199.99 | shipped    | 789 Pine St, Chicago |
|        5 | Ethan         | Keyboard    | 2025-08-03 08:20:00 |        5 |   49.99 | processing | NULL                 |
|        6 | Fiona         | Mouse       | 2025-08-04 10:30:00 |        4 |   25.00 | shipped    | 321 Oak St, Houston  |
+----------+---------------+-------------+--------------------+----------+---------+------------+----------------------+
```

LIKE

1. **Write a query to find all customers whose names end with 'a'. select customer name from orders where customer name like ‘%a’;**

```
mysql> select * from orders where customer_name like '%a';
+----------+---------------+-------------+--------------------+----------+--------+---------+----------------------+
| order_id | customer_name | product_name | order_date         | quantity | price  |status   |shipping_address      |
+----------+---------------+-------------+--------------------+----------+--------+---------+----------------------+
|        4 | Diana         | Monitor     | 2025-08-07 15:40:00 |        3 | 199.99 | shipped | 789 Pine St, Chicago |
|        6 | Fiona         | Mouse       | 2025-08-04 10:30:00 |        4 |  25.00 | shipped | 321 Oak St, Houston  |
+----------+---------------+-------------+--------------------+----------+--------+---------+----------------------+
```

2. **Write a query to find orders where the product name contains the word 'Phone'. Select product name from orders where product name like ’phone’;**

```
mysql> select * from orders where product_name like '%phone%';
```

+----------+---------------+-------------+--------------------+----------+--------+---------+------------------+

| order_id | customer_name | product_name | order_date | quantity | price |status |shipping_address |

+----------+---------------+-------------+--------------------+----------+--------+---------+------------------+

| 2 | Bob | Phone | 2025-08-06 14:10:00 | 2 | 650.50 | pending | 456 Elm St, LA |

+----------+---------------+-------------+--------------------+----------+--------+---------+------------------+

3. **Write a query to find orders where the product name is exactly 5 characters long.**

   Select product name from orders where product name like '_____';

   mysql> select * from orders where length(product_name) =5;

+----------+---------------+-------------+--------------------+----------+--------+---------+---------------------+

| order_id | customer_name | product_name | order_date | quantity | price |status |shipping_address |

+----------+---------------+-------------+--------------------+----------+--------+---------+---------------------+

| 2 | Bob | Phone | 2025-08-06 14:10:00 | 2 | 650.50 | pending | 456 Elm St, LA |

| 6 | Fiona | Mouse | 2025-08-04 10:30:00 | 4 | 25.00 | shipped | 321 Oak St, Houston |

+----------+---------------+-------------+--------------------+----------+--------+---------+---------------------+

AS (Alias)

5. **Write a query to show customer name as Buyer and price as Unit_Price.**

   mysql> select customer_name as Buyer, price as Unit_price from orders;

+---------+------------+

| Buyer | Unit_price |

+---------+------------+

| Alice | 1200.00 |

| Bob | 650.50 |

| Charlie | 300.00 |

| Diana | 199.99 |

| Ethan | 49.99 |

| Fiona | 25.00 |

+---------+------------+

6. **Write a query to show order ID and total amount (quantity × price) as Total_Cost.**

```
mysql> select order_id,(quantity*price) as Total_Cost from orders;
+----------+------------+
| order_id | Total_Cost |
+----------+------------+
|        1 |    1200.00 |
|        2 |    1301.00 |
|        3 |     300.00 |
|        4 |     599.97 |
|        5 |     249.95 |
|        6 |     100.00 |
+----------+------------+
```

NOT

7. **Write a query to find all orders not placed by 'Bob'.**

```
mysql> select order_id,(quantity*price) as Total_Cost from orders;
+----------+------------+
| order_id | Total_Cost |
+----------+------------+
|        1 |    1200.00 |
|        2 |    1301.00 |
|        3 |     300.00 |
|        4 |     599.97 |
|        5 |     249.95 |
|        6 |     100.00 |
+----------+------------+
```

8. **Write a query to find orders where status is not 'shipped'.**

```
mysql> select * from orders where status!='shipped';
+----------+---------------+--------------+---------------------+----------+--------+------------+------------------+
| order_id | customer_name | product_name | order_date          | quantity | price  | status     | shipping_address |
```

+----------+---------------+-------------+---------------------+----------+--------+------------+------------------+

| 2 | Bob | Phone | 2025-08-06 14:10:00 | 2 | 650.50 | pending | 456 Elm St, LA |

| 3 | Charlie | Tablet | 2025-08-05 11:25:00 | 1 | 300.00 | cancelled | NULL |

| 5 | Ethan | Keyboard | 2025-08-03 08:20:00 | 5 | 49.99 | processing | NULL |

+----------+---------------+-------------+---------------------+----------+--------+------------+------------------+

AND

9. **Write a query to find orders with quantity greater than 2 and price less than 500.**

mysql> select * from orders where quantity>2 and price<500;

+----------+---------------+-------------+---------------------+----------+--------+------------+---------------------+

| order_id | customer_name | product_name | order_date | quantity | price | status | shipping_address |

+----------+---------------+-------------+---------------------+----------+--------+------------+---------------------+

| 4 | Diana | Monitor | 2025-08-07 15:40:00 | 3 | 199.99 | shipped | 789 Pine St, Chicago |

| 5 | Ethan | Keyboard | 2025-08-03 08:20:00 | 5 | 49.99 | processing | NULL |

| 6 | Fiona | Mouse | 2025-08-04 10:30:00 | 4 | 25.00 | shipped | 321 Oak St, Houston |

+----------+---------------+-------------+---------------------+----------+--------+------------+---------------------+

10. **Write a query to find orders placed by 'Alice' and status is 'shipped'.**

mysql> select* from orders where customer_name like 'alice' and status like 'shipped';

+----------+---------------+-------------+---------------------+----------+---------+---------+----------------------+

| order_id | customer_name | product_name | order_date | quantity | price | status |shipping_address |

+----------+---------------+-------------+---------------------+----------+---------+---------+----------------------+

| 1 | Alice | Laptop | 2025-08-07 09:00:00 | 1 | 1200.00 | shipped | 123 Main St, New York |

+----------+---------------+-------------+---------------------+----------+---------+---------+----------------------+

OR

11. **Write a query to find orders where status is 'pending' or 'processing'.**

mysql> select * from orders where status like 'pending' or 'processing';

+----------+---------------+-------------+--------------------+----------+--------+---------+-----------------+

| order_id | customer_name | product_name | order_date | quantity | price |status |shipping_address |

+----------+---------------+-------------+--------------------+----------+--------+---------+-----------------+

| 2 | Bob | Phone | 2025-08-06 14:10:00 | 2 | 650.50 | pending | 456 Elm St, LA |

+----------+---------------+-------------+--------------------+----------+--------+---------+-----------------+

12. **Write a query to find orders made by 'Alice' or 'Diana'.**

mysql> select * from orders where customer_name like 'alice' or 'diana';

+----------+---------------+-------------+--------------------+----------+--------+---------+----------------------+

| order_id | customer_name | product_name | order_date | quantity | price | status |shipping_address |

+----------+---------------+-------------+--------------------+----------+--------+---------+----------------------+

| 1 | Alice | Laptop | 2025-08-07 09:00:00 | 1 | 1200.00 | shipped | 123 Main St, New York |

+----------+---------------+-------------+--------------------+----------+--------+---------+----------------------+

IN

13. **Write a query to find orders for products in the categories: 'Laptop', 'Phone', or 'Tablet'.**

mysql> select * from orders where product_name in('laptop','phone','tablet');

+----------+---------------+-------------+--------------------+----------+--------+-----------+----------------------+

| order_id | customer_name | product_name | order_date | quantity | price | status | shipping_address |

+----------+---------------+-------------+--------------------+----------+--------+-----------+----------------------+

| 1 | Alice | Laptop | 2025-08-07 09:00:00 | 1 | 1200.00 | shipped | 123 Main St, New York |

| 2 | Bob | Phone | 2025-08-06 14:10:00 | 2 | 650.50 | pending | 456 Elm St, LA |

| 3 | Charlie | Tablet | 2025-08-05 11:25:00 | 1 | 300.00 | cancelled | NULL |

+----------+---------------+-------------+--------------------+----------+--------+-----------+----------------------+

14. **Write a query to find orders placed by customers in a given list: 'Alice', 'Bob', 'Charlie'.**

mysql> select * from orders where product_name in('laptop','phone','tablet');

+----------+---------------+-------------+--------------------+----------+---------+-----------+----------------------+

| order_id | customer_name | product_name | order_date | quantity | price | status | shipping_address |

+----------+---------------+-------------+--------------------+----------+---------+-----------+----------------------+

| 1 | Alice | Laptop | 2025-08-07 09:00:00 | 1 | 1200.00 | shipped | 123 Main St, New York |

| 2 | Bob | Phone | 2025-08-06 14:10:00 | 2 | 650.50 | pending | 456 Elm St, LA |

| 3 | Charlie | Tablet | 2025-08-05 11:25:00 | 1 | 300.00 | cancelled | NULL |

+----------+---------------+-------------+--------------------+----------+---------+-----------+----------------------+

IS NULL / IS NOT NULL

15. **Write a query to find orders where the shipping address is missing.**

mysql> select * from orders where shipping_address is null;

+----------+---------------+-------------+--------------------+----------+--------+------------+-----------------+

| order_id | customer_name | product_name | order_date | quantity | price | status | shipping_address |

+----------+---------------+-------------+--------------------+----------+--------+------------+-----------------+

| 3 | Charlie | Tablet | 2025-08-05 11:25:00 | 1 | 300.00 | cancelled | NULL |

| 5 | Ethan | Keyboard | 2025-08-03 08:20:00 | 5 | 49.99 | processing | NULL |

+----------+---------------+-------------+--------------------+----------+--------+------------+-----------------+

16. **Write a query to find orders where the shipping address is provided.**

mysql> select * from orders where shipping_address != 'null';

+----------+---------------+-------------+--------------------+----------+---------+---------+----------------------+

| order_id | customer_name | product_name | order_date | quantity | price | status |shipping_address |

+----------+---------------+-------------+--------------------+----------+---------+---------+----------------------+

| 1 | Alice | Laptop | 2025-08-07 09:00:00 | 1 | 1200.00 | shipped | 123 Main St, New York |

| 2 | Bob | Phone | 2025-08-06 14:10:00 | 2 | 650.50 | pending | 456 Elm St, LA |

| 4 | Diana | Monitor | 2025-08-07 15:40:00 | 3 | 199.99 | shipped | 789 Pine St, Chicago |

| 6 | Fiona | Mouse | 2025-08-04 10:30:00 | 4 | 25.00 | shipped | 321 Oak St, Houston |

+----------+---------------+-------------+---------------------+----------+---------+---------+----------------------+

FUNCTION-BASED

17. **Write a query to find customers whose name is longer than 5 characters.**

```
mysql> select * from orders where length(customer_name)>5;
+----------+---------------+-------------+---------------------+----------+--------+-----------+------------------+
| order_id | customer_name | product_name | order_date          | quantity | price  | status    | shipping_address |
+----------+---------------+-------------+---------------------+----------+--------+-----------+------------------+
|        3 | Charlie       | Tablet      | 2025-08-05 11:25:00 |        1 | 300.00 | cancelled | NULL             |
+----------+---------------+-------------+---------------------+----------+--------+-----------+------------------+
```

18. **Write a query to show all customer names in uppercase.**

```
mysql> select upper(customer_name) from orders;
+----------------------+
| upper(customer_name) |
+----------------------+
| ALICE                |
| BOB                  |
| CHARLIE              |
| DIANA                |
| ETHAN                |
| FIONA                |
+----------------------+
```

19. **Write a query to find orders where the product name is all lowercase.**

```
mysql> select lower(customer_name) from orders;
+----------------------+
| lower(customer_name) |
+----------------------+
| alice                |
| bob                  |
| charlie              |
```

| diana |

| ethan |

| fiona |

+----------------------+

6 rows in set (0.02 sec)

20. **Write a query to return the first 3 letters of each product name with an alias Short_Name.**

mysql> select substring(product_name,1,3) as Short_Name from orders;

+------------+

| Short_Name |

+------------+

| Lap |

| Pho |

| Tab |

| Mon |

| Key |

| Mou |

+------------+