

Example data

We'll use this data in all the examples below.

customers table:

customer_id	customer_name	country
1	Alice	USA
2	Carlos	Mexico
3	Bob	Canada
4	Diana	France

orders table:

order_id	customer_id	amount	order_date
101	1	250	2025-08-05
102	2	75	2025-08-06
103	3	120	2025-08-07
104	4	45	2025-08-08

Getting data

Use **SELECT** and **FROM** to choose data from a table.

Get all columns from a table

```
SELECT customer_name
FROM customers;
```

Get specific columns from a table

```
SELECT *
FROM orders;
```

Get unique values from a column

```
SELECT DISTINCT country
FROM customers;
```

Operators in SQL

- Comparison
- `=` → equal to
 - `<>` → not equal
 - `>` / `<` → greater / less than
 - `>=` / `<=` → greater or equal / less or equal

- Logical
- `AND` → both true
 - `OR` → either true
 - `NOT` → exclude rows

Filtering data

Use **WHERE** to filter rows by conditions.

Filter rows above a certain amount

```
SELECT *
FROM orders
WHERE amount > 100;
```

Filter rows matching multiple conditions

```
SELECT *
FROM orders
WHERE country = 'USA' AND amount > 100;
```

Filter rows matching either condition

```
SELECT *
FROM orders
WHERE country = 'USA' OR country = 'Canada';
```

Filter rows by a list of values

```
SELECT *
FROM customers
WHERE country IN ('USA', 'Canada');
```

Exclude rows with specific values

```
SELECT *
FROM customers
WHERE country NOT IN ('USA');
```

Filter rows within a range

```
SELECT *
FROM orders
WHERE amount BETWEEN 50 AND 200;
```

Filter rows by text patterns

```
SELECT *
FROM customers
WHERE customer_name LIKE 'A%';
```

Filter rows by date

```
SELECT *
FROM orders
WHERE order_date >= '2024-02-01';
```

Agregating data

Use **COUNT**, **SUM**, **AVG**, and **GROUP BY** to summarize data.

Count total rows

```
SELECT COUNT(*)
FROM orders;
```

Count distinct values

```
SELECT COUNT(DISTINCT customer_id)
FROM orders;
```

Calculate total value

```
SELECT SUM(amount)
FROM orders;
```

Calculate average value

```
SELECT AVG(amount)
FROM orders;
```

Group rows by a column

```
SELECT country, COUNT(*)
FROM customers
GROUP BY country;
```

Group rows and calculate totals

```
SELECT customer_id, SUM(amount)
FROM orders
GROUP BY customer_id;
```

Filter grouped results

```
SELECT customer_id, SUM(amount)
FROM orders
GROUP BY customer_id
HAVING SUM(amount) > 200;
```

Use **CASE** to create if/then rules in your query:

```
SELECT order_id, amount,
CASE
    WHEN amount > 100 THEN 'High'
    ELSE 'Low'
END
FROM orders;
```

Organizing results

Sort results with **ORDER BY**, restrict them with **LIMIT**.

Sort rows in ascending order

```
SELECT *
FROM orders
ORDER BY order_date;
```

Sort rows in descending order

```
SELECT *
FROM orders
ORDER BY amount DESC;
```

Restrict the number of rows returned

```
SELECT *
FROM customers
LIMIT 3;
```

Sort rows and return only the top results

```
SELECT order_id, amount
FROM orders
ORDER BY amount DESC
LIMIT 3;
```

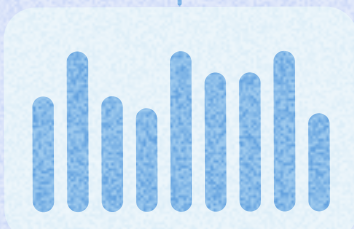
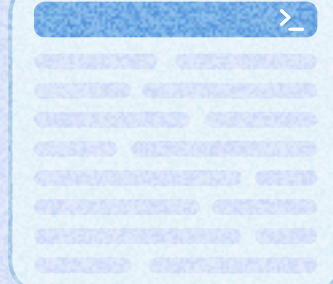
Sort rows by multiple columns

```
SELECT customer_name, country
FROM customers
ORDER BY country ASC,
        customer_name ASC;
```

Skip rows with **OFFSET**

```
SELECT *
FROM orders
ORDER BY order_date
LIMIT 3 OFFSET 2;
```

Write SQL in Metabase and see results as charts.



metabase.com