

SQL Introduction

SQL SELECT (I)

SQL SELECT (II)

SQL ORDER BY

SQL GROUP BY

SQL LIKE

SQL Wildcards

SQL UNION

SQL Subquery

SQL ANY and ALL

SQL CASE

SQL HAVING

SQL EXISTS

SQL JOIN

SQL DATABASE & TABLE

SQL Insert, Update and Delete

SQL Constraints

SQL Additional Topics

#### Related Topics

- SQL SELECT INTO Statement
- SQL GROUP BY
- SQL Stored Procedures
- SQL INSERT INTO SELECT Statement
- SQL HAVING Clause
- SQL IN Operator

## SQL CASE

In this tutorial, we'll learn about the CASE statement in SQL and how to use them with examples.

The `CASE` statement in SQL is used to check conditions and perform tasks on each row while selecting data. For example,

```
SELECT customer_id, first_name,
CASE
  WHEN age >= 18 THEN 'Allowed'
END AS can_vote
FROM Customers;
```

Run Code

Here, the SQL command checks each row with the given case.

If `age` is greater than or equal to **18**, the result set contains

- columns with `customer_id` and `first_name` with their values
- Allowed** is returned as a `can_vote` column.

Table: Customers

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

```
SELECT customer_id, first_name
CASE
  WHEN age >= 18 THEN 'Allowed'
END AS can_vote
FROM Customers;
```

customer_id	first_name	can_vote
1	John	Allowed
2	Robert	Allowed
3	David	Allowed
4	John	Allowed
5	Betty	Allowed

Example: CASE in SQL

**Note:** The syntax of `CASE` always starts with the `CASE` keyword and ends with the `END` keyword followed by column name alias.

### Example Two: SQL CASE Statement

Let's take a look at another example where we want to provide a **10% discount** on each order for a Christmas sale **if the amount is more than 400**.

```
SELECT order_id, customer_id,
CASE
  WHEN amount >= 400 THEN (amount - amount * 10/100)
END AS offer_price
FROM Orders;
```

Run Code

Here, the SQL command checks if the `amount` is greater than or equal to **400**. If this condition is satisfied, a new column `offer_price` will contain the values that's equal to `amount - amount*10/100`.

### Multiple Cases

It is also possible to stack multiple conditions inside the `CASE` clause.

```
SELECT customer_id, first_name,
CASE
  WHEN country = 'USA' THEN 'United States of America'
  WHEN country = 'UK' THEN 'United Kingdom'
END AS country_name
FROM Customers;
```

Run Code

Here, the result set will contain a column named `country_name` along with `customer_id` and `first_name` columns.

The value of `country_name` will be **United States of America** if the `country` is equal to **USA**.

Similarly, the value of `country_name` will be **United Kingdom** if the `country` is equal to **UK**.

### CASE With ELSE in SQL

A `CASE` statement can have an optional `ELSE` clause. The `ELSE` clause is executed if none of the conditions in the `CASE` statement is matched. For example,

```
SELECT customer_id, first_name,
CASE
  WHEN country = 'USA' THEN 'United States of America'
  WHEN country = 'UK' THEN 'United Kingdom'
  ELSE 'Unknown Country'
END AS country_name
FROM Customers;
```

Run Code

Here, the result set will contain a field named `country_name` along with `customer_id` and `first_name`.

The value of the `country_name` will be:

- United States of America** if the `country` is **USA**
- United Kingdom** if the `country` is **UK**
- Unknown Country** if the `country` is neither **USA** nor **UK** (because of the `ELSE` clause).

Table: Customers

customer_id	first_name	last_name	age	country
1	John	Doe	31	USA
2	Robert	Luna	22	USA
3	David	Robinson	22	UK
4	John	Reinhardt	25	UK
5	Betty	Doe	28	UAE

```
SELECT customer_id, first_name
CASE
  WHEN country = 'USA' THEN 'United States of America'
  WHEN country = 'UK' THEN 'United Kingdom'
  ELSE 'Unknown Country'
END AS country_name
FROM Customers;
```

customer_id	first_name	country_name
1	John	United States of America
2	Robert	United States of America
3	David	United Kingdom
4	John	United Kingdom
5	Betty	Unknown Country

Example: CASE With ELSE in SQL

Previous Tutorial:  
**SQL ANY and ALL**

Next Tutorial:  
**SQL HAVING**

Did you find this article helpful?



#### Related Tutorials

- Programming

SQL GROUP BY
- Programming

SQL SELECT INTO Statement
- Programming

SQL Stored Procedures
- Programming

SQL INSERT INTO SELECT Statement

Programiz



#### Tutorials

- Python 3 Tutorial
- JavaScript Tutorial
- SQL Tutorial
- C Tutorial
- Java Tutorial
- Kotlin Tutorial
- C++ Tutorial
- Swift Tutorial
- C# Tutorial
- Go Tutorial
- DSA Tutorial

#### Examples

- Python Examples
- JavaScript Examples
- C Examples
- Java Examples
- Kotlin Examples
- C++ Examples

#### Company

- About
- Advertising
- Privacy Policy
- Terms & Conditions
- Contact
- Blog
- Youtube

#### Apps

- Learn Python
- Learn C Programming
- Learn Java