

case

Description

Returns a value based on the first condition that evaluates to `true`.

The `case` statement allows you to perform multiple conditional checks against a value. Each condition must resolve to a boolean expression, and the first condition that matches determines the result.

NOTE

Like all case statements, `case` returns the **first matching value**. The order of clauses is important.

Syntax

```
case {  
  condition1 -> value1,  
  condition2 -> value2,  
  ...  
  conditionN -> valueN,  
  _          -> default  
}
```

Arguments

Name	Type	Required	Description
condition	boolean	true	A condition to evaluate
value	any	true	The value to return if the condition is true
_	any	false	Default value if no conditions match

Example

Use case: Map server IPs to owners by subnet

Suppose you want to enrich logs with the owner of each server based on its IP address. This can help route incidents to the right team.

You can use the `ipInSubnet` function inside a `case` statement to determine ownership. If no subnet matches, a default owner is assigned.

Example data

```
{
  "server_ip": "10.0.0.1"
},
{
  "server_ip": "165.0.0.1"
},
{
  "server_ip": "333.4.5.6"
}
```

Example query

```
create server_owner from
case {
  ipInSubnet(server_ip, '10.0.0.0/8')    -> 'Chris',
  ipInSubnet(server_ip, '165.0.0.0/8')   -> 'George',
  ipInSubnet(server_ip, '333.4.0.0/16')  -> 'Maya',
  _                                       -> 'DevOps Team'
}
```

Example output

```
{
  "server_ip": "10.0.0.1",
  "server_owner": "Chris"
},
{
  "server_ip": "165.0.0.1",
  "server_owner": "George"
},
{
  "server_ip": "333.4.5.6",
  "server_owner": "DevOps Team"
}
```

```
"server_owner": "Maya"  
}
```

🕒 Last updated: October 19, 2025

Was this helpful?

Leave your feedback here.

☐ **Yes** ☐ **No**

Send

© 2025 Coralogix. All rights reserved.

Generated on: November 17, 2025

Source: <https://coralogix.com/docs/dataprime/language-reference/functions-reference/cases/case/>