Logical Operators

So far, we've used comparison operators to filter the data. But in real-world scenarios, we often have to retrieve the data using several conditions at once. For example, in the case of e-commerce platforms, users often search for something like:
Get shoes from the Puma brand, which have ratings greater than 4.0 and price less than 5000.

With logical operators, we can perform queries based on multiple conditions. Let's learn how with the following database.

Database

The database contains a

`product` table that stores the data of products like name, category, price, brand and rating. You can check the schema and data of `product` table in the code playground.

AND, OR, NOT

Operator	Description
AND	Used to fetch rows that satisfy two or more conditions.
OR	Used to fetch rows that satisfy at least one of the given conditions.
NOT	Used to negate a condition in the `WHERE` clause.

Syntax

SQL

```sql
SELECT
  *
FROM
  table_name
WHERE
  condition1
  operator condition2
  operator condition3
  ...;
```

Examples

1. Get all the details of the products whose
    - `category` is "Clothing" *and*
    - `price` less than or equal to 1000 from the `product` table.

SQL

```sql
SELECT
  *
FROM
  product
WHERE
  category = "Clothing"
  AND price <= 1000;
```

Output

name	category	price	brand	rating
Blue Shirt	Clothing	750	Denim	3.8
Blue Jeans	Clothing	800	Puma	3.6
Black Jeans	Clothing	750	Denim	4.5
...	...	...	...	...

2. *Ignore* all the products with `name` containing "Cake" from the list of products.

SQL

```sql
SELECT
  *
FROM
  product
WHERE
  NOT name LIKE "%Cake%";
```

Output

name	category	price	brand	rating
Blue Shirt	Clothing	750	Denim	3.8
Blue Jeans	Clothing	800	Puma	3.6
Black Jeans	Clothing	750	Denim	4.5

name	category	price	brand	rating
---	---	---	---	---

Try it Yourself!

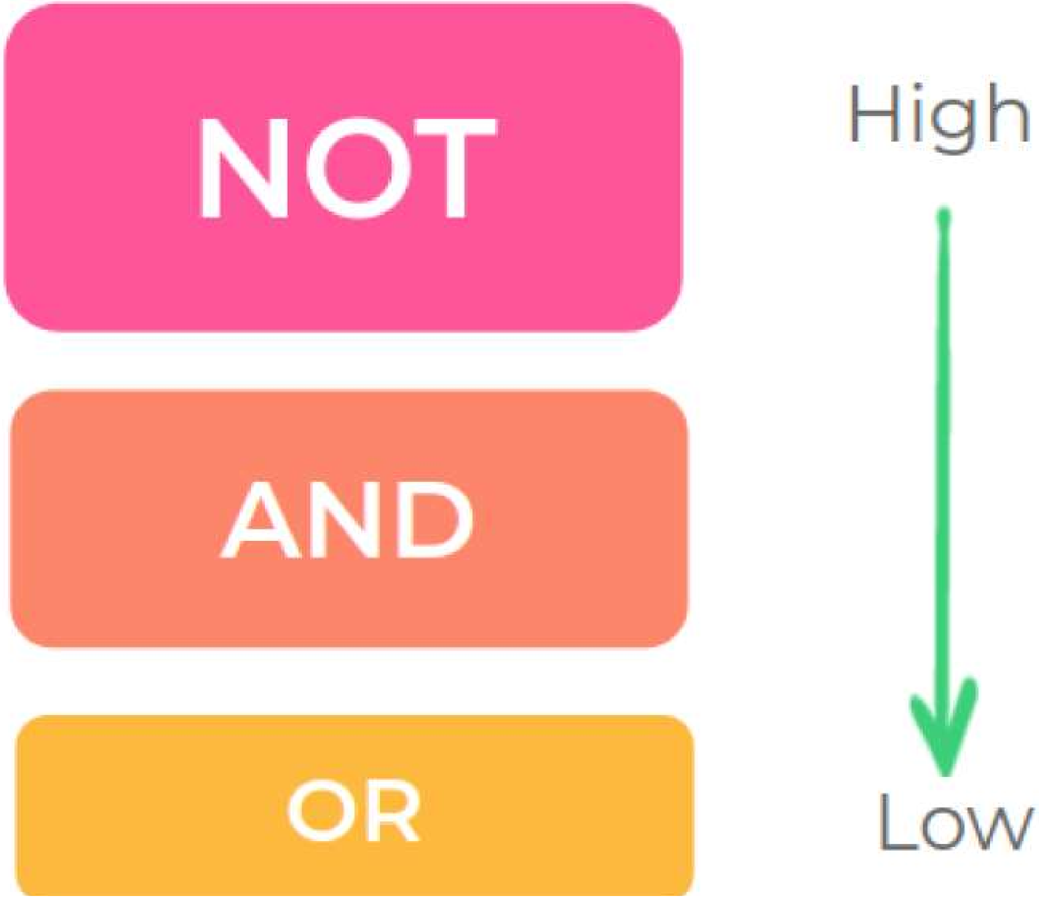
- Fetch all the products with `price` less than 20000 and `brand` is "Apple".
- Fetch all the products with `rating` greater than 4.0 or `brand` is "Britannia".
- Ignore all the products with `category` containing "Food" in `product` table.

Multiple Logical Operators

We can also use the combinations of logical operators to combine two or more conditions. These compound conditions enable us to fine-tune the data retrieval requirements.

Precedence

- When a query has multiple operators, operator precedence determines the sequence of operations.



Order of precedence:

- NOT
- AND
- OR

Example

Fetch the products that belong to

- *Redmi* `brand` and `rating` greater than *4* or
- the products from *OnePlus*

  `brand`

SQL

```sql
SELECT
  *
FROM
  product
WHERE
  brand = "Redmi"
  AND rating > 4
  OR brand = "OnePlus";
```

- In the above query,

  `AND` has the precedence over `OR` . So, the above query is equivalent to:

SQL

```sql
SELECT
    *
FROM
    product
WHERE
    (brand = "Redmi"
    AND rating > 4)
    OR brand = "OnePlus";
```

Quick Tip

It is suggested to always use parenthesis to ensure correctness while grouping the conditions.

Try it Yourself!

- Fetch all the products from "Clothing" category whose `name` does not contain "Jeans".
- Fetch all the products from "Puma" and "Denim" brands excluding the products with `name` containing "Shirts".
- Fetch all the products with `price` less than 100 or the products from "Food" category excluding the ones with `name` containing "Chocolate".