# Section 6: Filtering Data with SQL WHERE Clause

## Summary

### 6.2 Introduction to the WHERE Clause

In SQL, the SELECT statement helps you specify what data to retrieve, but it's the WHERE clause that lets you specify which exact rows to select. Think of it as a filter for your data – only allowing through the data that meets your specified criteria. Whether you're looking for specific customer details, transactions within a certain date range, or products of a certain type, the WHERE clause is your tool for honing in on that data.

### 6.3 Basic Filtering with WHERE

We'll start with the basics of the WHERE clause, understanding its syntax and how to implement simple filters. You'll learn how to filter rows based on single conditions, such as finding customers with a specific name or products with a certain price.

## 6.4 Using Operators in WHERE Clauses

To refine your filtering skills, we'll explore a variety of operators in SQL. These include equals (=), not equals (!=), greater than (>), less than (<), and more. Operators allow you to create more precise conditions for filtering data.

## 6.5 Logical Operators: AND, OR, NOT

Filtering often involves combining multiple conditions. We'll delve into logical operators like AND, OR, and NOT, which allow you to create complex conditions. You can find, for example, all customers who made a purchase in a specific month and spent more than a certain amount.

## 6.6 Working with NULL Values in WHERE Clauses

Handling NULL values is a common challenge in databases. We'll explore how to filter rows that contain NULL values or exclude them from your results. This skill is crucial for maintaining data accuracy.

## 6.7 Using LIKE and Wildcards for Pattern Matching

Sometimes, you need to search for patterns within text data. The LIKE operator, combined with wildcards, allows you to perform pattern matching. You can find, for instance, all products with names containing a specific keyword.

## 6.8 IN Operator for Multiple Values

The IN operator is a powerful tool for filtering rows that match multiple values. You can specify a list of values and retrieve rows where a column matches any of those values. This is handy when you want to filter data by multiple criteria simultaneously.

## 6.9 BETWEEN Operator for Range Filtering

The BETWEEN operator is ideal for filtering data within a specified range. You can find, for example, all orders placed between two dates or products with prices within a certain range.

# Code Breakdown

In this section, we will break down the SQL code used to construct various types of WHERE clause conditions. Understanding the code behind SQL WHERE clauses and operators is crucial for becoming proficient in data filtering. Let's explore the code associated with each aspect of SQL filtering:

## Basic WHERE Clause

```
SELECT column1, column2
FROM table_name
WHERE condition;
```

- `WHERE`: This clause allows you to specify conditions for filtering rows.
- `condition`: Replace this with the condition you want to apply, such as `column1 = 'value'`.

## Using Operators for Filtering

```
SELECT column1, column2
FROM table_name
WHERE column1 > value AND column2 = 'text';
```

- `AND`: Used to combine multiple conditions, ensuring both must be true for a row to be selected.

- `OR`: Allows you to select rows where at least one condition is true.
- `NOT`: Negates a condition, selecting rows where the condition is false.

## Handling NULL Values

```
SELECT column1, column2
FROM table_name
WHERE column1 IS NULL OR column2 IS NOT NULL;
```

- `IS NULL`: Used to filter rows with NULL values in a column.
- `IS NOT NULL`: Filters rows with non-NULL values.

## Using LIKE and Wildcards

```
SELECT column1, column2
FROM table_name
WHERE column1 LIKE 'pattern%';
```

- `LIKE`: Enables pattern matching using wildcards.
- `%`: Matches any sequence of characters.
- `_`: Matches a single character.

## IN Operator for Multiple Values

```
SELECT column1, column2
FROM table_name
WHERE column1 IN ('value1', 'value2', 'value3');
```

- `IN`: Filters rows where a column matches any of the specified values.

### BETWEEN Operator for Range Filtering

```
SELECT column1, column2
FROM table_name
WHERE column1 BETWEEN min_value AND max_value;
```

- `BETWEEN`: Filters rows within a specified range.

## Vocabulary

**WHERE Clause:** Part of an SQL query that specifies conditions for filtering rows.

**Operators:** Symbols or keywords used in SQL to perform comparisons and logical operations.

**Logical Operators:** Operators used to combine or negate conditions in SQL, including AND, OR, and NOT.

**Pattern Matching:** The process of searching for specific patterns within text data using wildcards.

**IN Operator:** An SQL operator used for filtering rows that match any of the specified values.

**BETWEEN Operator:** An SQL operator used for filtering rows within a specified range.

## Practice

Now that you've learned how to use the WHERE clause and various operators in SQL to filter data, it's time to practice your skills. These exercises will help reinforce your understanding and proficiency in data filtering.

You can use the database located at https://drive.google.com/file/d/1fEYPZN2EvrVMQMtXwOyP37wJg8KQDlLh/view?usp=sharing to complete these exercises.

## Exercise 1: Basic Filtering

Write SQL queries to perform the following basic filtering tasks:

1. Retrieve all customers with the name "John Smith" from the "customers" table.
2. Get a list of products with a price greater than $50 from the "products" table.
3. Fetch all orders placed on the date '2023-05-15' from the "orders" table.
4. Find employees who earn a salary less than or equal to $40,000 from the "employees" table.
5. Select books with the title containing the word "SQL" from the "books" table.

## Exercise 2: Using Operators

Practice using operators in WHERE clauses to create more precise conditions:

1. Retrieve all products with a price greater than or equal to 50.
2. Get a list of orders placed by the customer with the ID 102 or 105.
3. Fetch employees who joined the company after '2023-01-01' and earn a salary greater than $45,000.
4. Find products that are not in stock (quantity equals 0) or have a price less than $10.
5. Select books published in or after the year 2020 and not authored by "John Doe."

## Exercise 3: Logical Operators

Combine multiple conditions using logical operators to create complex filters:

1. Retrieve all customers who made a purchase in the month of May 2023 and spent more than $100.

2. Get a list of products that are either in the "Electronics" category or have a price greater than $100.
3. Fetch employees who earn a salary less than $40,000 and joined the company after '2023-02-01' or have the title "Manager."
4. Find products that are out of stock (quantity equals 0) and have a price less than 50.
5. Select orders placed in January 2023 by customers with names containing the word "Smith" or orders with a total amount exceeding $500.

## Exercise 4: Handling NULL Values

Practice filtering rows that contain NULL values or excluding them from results:

1. Retrieve customers who have provided an email address (email column is not NULL) from the "customers" table.
2. Get a list of products with a description (description column is not NULL) from the "products" table.
3. Fetch employees who do not have a manager (manager_id column is NULL) from the "employees" table.
4. Find orders placed by customers with a specified delivery address (delivery_address column is not NULL) from the "orders" table.
5. Select books with authors (author column is not NULL) from the "books" table.

## Exercise 5: Pattern Matching

Use the LIKE operator and wildcards to perform pattern matching on text data:

1. Retrieve all products with names containing the word "smart" from the "products" table.
2. Get a list of customers with names starting with "A" from the "customers" table.
3. Fetch employees with last names ending with "son" from the "employees" table.

4. Find orders with order numbers ending with "2023" from the "orders" table.
5. Select products with descriptions containing the phrase "high-quality" from the "products" table.

## Exercise 6: IN Operator

Practice using the IN operator to filter rows that match multiple values:

1. Retrieve orders placed by customers with IDs 101, 103, and 105 from the "orders" table.
2. Get a list of employees with job titles "Manager" or "Supervisor" from the "employees" table.
3. Fetch products with IDs 201, 205, 209, and 215 from the "products" table.
4. Find books with ISBNs '978-1234567890' and '978-9876543210' from the "books" table.
5. Select orders placed by customers with IDs 102, 104, 106, and 108 from the "orders" table.

## Exercise 7: BETWEEN Operator

Practice using the BETWEEN operator to filter data within a specified range:

1. Retrieve orders placed between '2023-03-01' and '2023-04-30' from the "orders" table.
2. Get a list of employees hired between '2023-05-01' and '2023-06-30' from the "employees" table.
3. Fetch products with prices between 50 from the "products" table.
4. Find books published between the years 2015 and 2020 from the "books" table.
5. Select orders with order amounts between 500 from the "orders" table.

# Exam Q & A

**Q1: What is the primary purpose of the WHERE clause in SQL?**

a) To specify the columns to retrieve from a table
b) To sort the query results
c) To perform aggregate calculations
d) To create new tables
**Answer: a) To specify the columns to retrieve from a table.**
*Explanation: The WHERE clause in SQL is used to specify conditions for filtering rows based on certain criteria, not for selecting columns.*

**Q2: Which SQL operator is used to combine multiple conditions in a WHERE clause, ensuring that all conditions must be true for a row to be selected?**
a) AND
b) OR
c) NOT
d) BETWEEN
**Answer: a) AND.**
*Explanation: The AND operator in SQL is used to combine multiple conditions, and all conditions must be true for a row to be selected.*

**Q3: What is the purpose of the LIKE operator in SQL?**
a) To perform aggregate calculations
b) To sort query results
c) To enable pattern matching using wildcards
d) To filter rows with NULL values
**Answer: c) To enable pattern matching using wildcards.**
*Explanation: The LIKE operator in SQL is used to search for specific patterns within text data using wildcards.*

**Q4: Which SQL operator is used to filter rows within a specified range of values?**
a) IN
b) BETWEEN
c) NOT
d) IS NULL
**Answer: b) BETWEEN.**
*Explanation: The BETWEEN operator in SQL is used to filter rows within a specified range of values.*

**Q5: What is the purpose of the IN operator in SQL?**
a) To perform aggregate calculations
b) To filter rows with NULL values
c) To filter rows where a column matches any of the specified values
d) To enable pattern matching using wildcards
**Answer: c) To filter rows where a column matches any of the specified values.**
*Explanation: The IN operator in SQL is used to filter rows where a column matches any of the specified values from a list.*