# SQL SELECT Statement

## The SQL SELECT Statement

**The SELECT statement is used to select data from a database.**

The data returned is stored in a result table, called the result-set.

### SELECT Syntax

|  |
| --- |
| **SELECT column1, column2, ... FROM table\_name;** |

Here, column1, column2, ... are the field names of the table you want to select data from.

**If you want to select all the fields available in the table, use the following syntax:**

|  |
| --- |
| SELECT \* FROM table\_name; |

## Demo Database

Below is a selection from the "Customers" table in the North wind sample database:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| CustomerID | CustomerName | ContactName | Address | City | PostalCode | Country |
| 1 | Alfreds Futterkiste | Maria Anders | Obere Str. 57 | Berlin | 12209 | Germany |
| 2 | Ana Trujillo Emparedados y helados | Ana Trujillo | Avda. de la Constitución 2222 | México D.F. | 05021 | Mexico |
| 3 | Antonio Moreno Taquería | Antonio Moreno | Mataderos 2312 | México D.F. | 05023 | Mexico |
| 4 | Around the Horn | Thomas Hardy | 120 Hanover Sq. | London | WA1 1DP | UK |
| 5 | Berglunds snabbköp | Christina Berglund | Berguvsvägen 8 | Luleå | S-958 22 | Sweden |

## SELECT Column Example

The following SQL statement selects the "CustomerName" and "City" columns from the "Customers" table:

Example

|  |
| --- |
| SELECT CustomerName, City FROM Customers; |

## Exercise

1.Insert the missing statement to get all the columns from the Customers table

 \* FROM Customers;

Answer: **select**

2. Write a statement that will select the City column from the Customers table.

   Customers;

Answer:

**Select city from customers;**

# SQL SELECT DISTINCT Statement

## The SQL SELECT DISTINCT Statement

The SELECT DISTINCT statement is used to return only distinct (different) values.

Inside a table, a column often contains many duplicate values; and sometimes you only want to list the different (distinct) values.

### SELECT DISTINCT Syntax

### SQL WHERE Clause

### The SQL WHERE Clause

**The WHERE clause is used to filter records.**

It is used to extract only those records that fulfill a specified condition.

### WHERE Syntax

|  |
| --- |
| SELECT column1, column2, ... FROM table\_name WHERE condition; |

**Note:** **The WHERE clause is not only used in SELECT statements, it is also used in UPDATE, DELETE, etc.!**

## WHERE Clause Example

**The following SQL statement selects all the customers from the country "chennai", in the "Customers" table:**

### Text Fields vs. Numeric Fields

**SQL requires single quotes around text values (most database systems will also allow double quotes).**

**However, numeric fields should not be enclosed in quotes:**

### Example

|  |
| --- |
| SELECT \* FROM Customers WHERE CustomerID=1; |

### Operators in The WHERE Clause

|  |  |
| --- | --- |
| Operator | Description |
| = | **Equal** |
| > | **Greater than** |
| < | **Less than** |
| >= | **Greater than or equal** |
| <= | **Less than or equal** |
| <> | **Not equal. Note: In some versions of SQL this operator may be written as !=** |
| BETWEEN | **Between a certain range** |

**The following operators can be used in the WHERE clause:**

### Exercise:

Select all records where the City column has the value "Berlin".

|  |
| --- |
| SELECT \* FROM Customers   1. WHERE CITY ‘chennai’   2. select\*from customers  WHERE postalcode ='600054'; Syntax |
| select\*from customers  WHERE postalcode ='600054'; --where | |
| SELECT \* FROM customers  WHERE price = 18; --Equal (=) | |
| SELECT \* FROM customers  WHERE Price > 30; --greater(>) | |
| SELECT \* FROM customers  WHERE Price < 30; --(<)lesser than | |
| SELECT \* FROM customers  WHERE Price >= 30; --Greater than or equal >= | |
| SELECT \* FROM customers  WHERE Price <= 30 --Less than or equal <= | |
| SELECT \* FROM customers  WHERE Price <> 18; --<> Not equal. Note: In some versions of SQL this  operator may be written as != | |
| SELECT \* FROM customers  WHERE Price BETWEEN 50 AND 60; -- Between a certain range(AND) | |

#### SQL AND, OR and NOT Operators

##### The SQL AND, OR and NOT Operators

The WHERE clause can be combined with AND, OR, and NOT operators.

The AND and OR operators are used to filter records based on more than one condition:

* The AND operator displays a record if all the conditions separated by AND are TRUE.
* The OR operator displays a record if any of the conditions separated by OR is TRUE.

The NOT operator displays a record if the condition(s) is NOT TRUE.

##### AND Syntax

|  |
| --- |
| SELECT column1, column2, ... FROM table\_name WHERE condition1 AND condition2 AND condition3 ...; |

##### OR Syntax

|  |
| --- |
| SELECT column1, column2, ... FROM table\_name WHERE condition1 OR condition2 OR condition3 ...; |

##### NOT Syntax

|  |
| --- |
| SELECT column1, column2, ... FROM table\_name WHERE NOT condition; |

##### Demo Database

The table below shows the complete "Customers" table from the Northwind sample database:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **CustomerID** | **CustomerName** | **ContactName** | **Address** | **City** |
| 1 | Alfreds Futterkiste | Maria Anders | Obere Str. 57 | Berlin |
| 2 | Ana Trujillo Emparedados y helados | Ana Trujillo | Avda. de la Constitución 2222 | México D.F. |
| 3 | Antonio Moreno Taquería | Antonio Moreno | Mataderos 2312 | México D.F. |

##### AND Example

The following SQL statement selects all fields from "Customers" where country is "Germany" AND city is "Berlin":

##### Example

|  |
| --- |
| SELECT \* FROM Customers WHERE Country='Germany' AND City='Berlin'; |

##### OR Example

The following SQL statement selects all fields from "Customers" where city is "Berlin" OR "München":

### Example

|  |
| --- |
| SELECT \* FROM Customers WHERE City='Berlin' OR City='München'; |

##### The following SQL statement selects all fields from "Customers" where country is "Germany" OR "Spain":

**Example**

|  |
| --- |
| SELECT \* FROM Customers WHERE Country='Germany' OR Country='Spain' |

##### NOT Example

The following SQL statement selects all fields from "Customers" where country is NOT "Germany":

|  |
| --- |
| SELECT \* FROM Customers WHERE NOT Country='Germany'; |

##### Combining AND, OR and NOT

You can also combine the AND, OR and NOT operators.

The following SQL statement selects all fields from "Customers" where country is "Germany" AND city must be "Berlin" OR "München" (use parenthesis to form complex expressions):

### Example

|  |
| --- |
| SELECT \* FROM Customers WHERE Country='Germany' AND (City='Berlin' OR City='München'); |

The following SQL statement selects all fields from "Customers" where country is NOT "Germany" and NOT "USA":

### Example

|  |
| --- |
| SELECT \* FROM Customers WHERE NOT Country='Germany' AND NOT Country='USA'; |

##### Exercise:

Select all records where the City column has the value 'Berlin' and the PostalCode column has the value 12209.

 \* FROM Customers

 City = 'Berlin'

  = 12209;

Answer : select where and postalcode

###### SQL ORDER BY KEYWORD

###### The SQL ORDER BY Keyword

The ORDER BY keyword is used to sort the result-set in ascending or descending order.

The ORDER BY keyword sorts the records in ascending order by default. To sort the records in descending order, use the DESC keyword.

###### ORDER BY Syntax

|  |
| --- |
| SELECT column1, column2, ... FROM table\_name ORDER BY column1, column2, ... ASC|DESC; |

###### Demo Database

Below is a selection from the "Customers" table in the Northwind sample database:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **CustomerID** | **CustomerName** | **ContactName** | **Address** | **City** | **PostalCode** | **Country** |
| 1 | Alfreds Futterkiste | Maria Anders | Obere Str. 57 | Berlin | 12209 | Germany |
| 2 | Ana Trujillo Emparedados y helados | Ana Trujillo | Avda. de la Constitución 2222 | México D.F. | 05021 | Mexico |
| 3 | Antonio Moreno Taquería | Antonio Moreno | Mataderos 2312 | México D.F. | 05023 | Mexico |
| 4 | Around the Horn | Thomas Hardy | 120 Hanover Sq. | London | WA1 1DP | UK |

###### ORDER BY DESC Example

The following SQL statement selects all customers from the "Customers" table, sorted DESCENDING by the "Country" column:

Example:

|  |
| --- |
| SELECT \* FROM Customers ORDER BY Country DESC; |

###### ORDER BY Several Columns Example

The following SQL statement selects all customers from the "Customers" table, sorted by the "Country" and the "CustomerName" column. This means that it orders by Country, but if some rows have the same Country, it orders them by CustomerName:

###### Example

|  |
| --- |
| SELECT \* FROM Customers ORDER BY Country, CustomerName; |

###### DER BY Several Columns Example 2

The following SQL statement selects all customers from the "Customers" table, sorted ascending by the "Country" and descending by the "CustomerName" column:

###### Example

|  |
| --- |
| SELECT \* FROM Customers ORDER BY Country ASC, CustomerName DESC; |

###### Exercise:

1. Select all records from the Customers table, sort the result alphabetically by the column City.

SELECT \* FROM Customers

 ;

Answer: Order by city;

2. Select all records from the Customers table, sort the result reversed alphabetically by the column City.

SELECT \* FROM Customers

  ;

Answer: Order by city DESC;

3. Select all records from the Customers table, sort the result alphabetically, first by the column Country, then, by the column City.

SELECT \* FROM Customers

  ;

Answer: ORDER BY Country,City;