

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.

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;
```

Example 1:

Below is the selection of all data from "Student" table from a "University" database.

✓ Showing rows 0 - 4 (5 total, Query took 0.0014 seconds.)

SELECT * FROM student

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: | Sort by key:

+ Options

				Id	Firstname	Lastname	Email	DateOfBirth	reg_date
<input type="checkbox"/>				1	Afaq	Shah	afaq@gmail.com	1998-03-12	2020-03-18 00:00:00
<input type="checkbox"/>				2	Asad	Khan	asad@gmail.com	1998-03-12	2020-03-18 00:00:00
<input type="checkbox"/>				3	Fahad	Iqbal	fahad@gmail.com	1991-03-15	2019-02-13 14:15:21
<input type="checkbox"/>				4	Zahid	Irfan	zahid@gmail.com	1995-05-23	2018-11-08 14:17:00
<input type="checkbox"/>				5	Saad	Akhtar	saad@gmail.com	1994-01-28	2021-03-25 14:20:14

Example 2:

The following SQL statement selects the "First Name" and "Email" columns from the "Student" table:

Showing rows 0 - 4 (5 total, Query took 0.0014 seconds.)

`SELECT Firstname, Email FROM student`

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

				Firstname	Email
<input type="checkbox"/>	Edit	Copy	Delete	Afaq	afaq@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Asad	asad@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Fahad	fahad@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Zahid	zahid@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Saad	saad@gmail.com

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.

Syntax:

```
SELECT DISTINCT column1, column2, ...
FROM table_name;
```

Example 1:

The following SQL statement selects all (including the duplicates) values from the "First Name" and "Email" column in the "Student" table:

Showing rows 0 - 5 (6 total, Query took 0.0016 seconds.)

`select Firstname, Email from student`

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

				Firstname	Email
<input type="checkbox"/>	Edit	Copy	Delete	Afaq	afaq@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Asad	asad@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Fahad	fahad@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Zahid	zahid@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Saad	saad@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Fahad	fahad@gmail.com

Now, let us use the SELECT DISTINCT statement and see the result.

Example 2:

The following SQL statement selects only the DISTINCT values from the "First Name" and "Email" column in the "Student" table:

✓ Showing rows 0 - 4 (5 total, Query took 0.0017 seconds.)

```
select distinct Firstname, Email from student
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: | Sort by key: None

+ Options

				▼ Firstname	Email
<input type="checkbox"/>	Edit	Copy	Delete	Afaq	afaq@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Asad	asad@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Fahad	fahad@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Zahid	zahid@gmail.com
<input type="checkbox"/>	Edit	Copy	Delete	Saad	saad@gmail.com

Example 3:

The following SQL statement lists the number of different (distinct) students First Names:

Your SQL query has been executed successfully.

```
SELECT COUNT(DISTINCT Firstname) FROM student
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#)

+ Options

COUNT(DISTINCT Firstname)

5

SQL Where

The WHERE clause is used to filter records.

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

Syntax:

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

Example:

The following SQL statement selects all the Students from the city "Lahore", in the "Student" table:

Showing rows 0 - 2 (3 total, Query took 0.0025 seconds.)

```
Select * from student where city = 'Lahore'
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

					Id	Firstname	Lastname	Email	DateOfBirth	reg_date	City
<input type="checkbox"/>					1	Afaq	Shah	afaq@gmail.com	1998-03-12	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					4	Zahid	Irfan	zahid@gmail.com	1995-05-23	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					6	Fahad	Khan	fahad@gmail.com	1994-01-28	2021-03-25 14:45:36	Lahore

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:

Showing rows 0 - 0 (1 total, Query took 0.0016 seconds.)

```
SELECT * FROM student WHERE Id=5
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table

+ Options

					Id	Firstname	Lastname	Email	DateOfBirth	reg_date	City
<input type="checkbox"/>					5	Saad	Akhtar	saad@gmail.com	1994-01-28	2021-03-25 14:45:36	Peshawar

Operators in the WHERE Clause

S No.	Operator	Description
1	=	Equal
2	>	Greater than
3	<	Less than
4	>=	Greater than or equal
5	<=	Less than or equal
6	<>	Not equal. Note: In some versions of SQL this operator may be written as !=
7	BETWEEN	Between a certain range
8	LIKE	Search for a pattern
9	IN	To specify multiple possible values for a column

- Less than Operator:

✓ Showing rows 0 - 3 (4 total, Query took 0.0018 seconds.)

```
SELECT * FROM student WHERE Id<5
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

					Id	Firstname	Lastname	Email	DateOfBirth	reg_date	City
<input type="checkbox"/>					1	Afaq	Shah	afaq@gmail.com	1998-03-12	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					2	Asad	Khan	asad@gmail.com	1998-03-12	2021-03-25 14:45:36	Peshawar
<input type="checkbox"/>					3	Fahad	Iqbal	fahad@gmail.com	1991-03-15	2021-03-25 14:45:36	Karachi
<input type="checkbox"/>					4	Zahid	Irfan	zahid@gmail.com	1995-05-23	2021-03-25 14:45:36	Lahore

- LIKE Operator:

✓ Showing rows 0 - 1 (2 total, Query took 0.0018 seconds.)

```
SELECT * FROM student WHERE Firstname LIKE 'a%'
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

					Id	Firstname	Lastname	Email	DateOfBirth	reg_date	City
<input type="checkbox"/>					1	Afaq	Shah	afaq@gmail.com	1998-03-12	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					2	Asad	Khan	asad@gmail.com	1998-03-12	2021-03-25 14:45:36	Peshawar

(Try more operators)

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.

Syntaxes:

```
AND: SELECT column1, column2, ...  
FROM table_name  
WHERE condition1 AND condition2 AND condition3 ...;
```

```
OR: SELECT column1, column2, ...  
FROM table_name  
WHERE condition1 OR condition2 OR condition3 ...;
```

```
NOT: SELECT column1, column2, ...  
FROM table_name  
WHERE NOT condition;
```

Example 1(AND):

Showing rows 0 - 0 (1 total, Query took 0.0017 seconds.)

```
SELECT * FROM student WHERE Firstname='Fahad' AND city='Lahore'
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows:

+ Options

	Id	Firstname	Lastname	Email	DateOfBirth	reg_date	City
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	Fahad	Khan	fahad@gmail.com	1994-01-28	2021-03-25 14:45:36	Lahore

Example 2 (OR):

Showing rows 0 - 3 (4 total, Query took 0.0018 seconds.)

```
SELECT * FROM student WHERE Firstname = 'Fahad' OR City='Lahore'
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

					Id	Firstname	Lastname	Email	DateOfBirth	reg_date	City
<input type="checkbox"/>					1	Afaq	Shah	afaq@gmail.com	1998-03-12	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					3	Fahad	Iqbal	fahad@gmail.com	1991-03-15	2021-03-25 14:45:36	Karachi
<input type="checkbox"/>					4	Zahid	Irfan	zahid@gmail.com	1995-05-23	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					6	Fahad	Khan	fahad@gmail.com	1994-01-28	2021-03-25 14:45:36	Lahore

Example 3 (NOT):

Showing rows 0 - 3 (4 total, Query took 0.0018 seconds.)

```
SELECT * FROM student WHERE NOT city='Peshawar'
```

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

+ Options

					Id	Firstname	Lastname	Email	DateOfBirth	reg_date	City
<input type="checkbox"/>					1	Afaq	Shah	afaq@gmail.com	1998-03-12	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					3	Fahad	Iqbal	fahad@gmail.com	1991-03-15	2021-03-25 14:45:36	Karachi
<input type="checkbox"/>					4	Zahid	Irfan	zahid@gmail.com	1995-05-23	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					6	Fahad	Khan	fahad@gmail.com	1994-01-28	2021-03-25 14:45:36	Lahore

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.

Syntax:

```
SELECT column1, column2, ...  
FROM table_name  
ORDER BY column1, column2, ... ASC|DESC;
```

Example 1:

✓ Showing rows 0 - 5 (6 total, Query took 0.0017 seconds.) [City: KARACHI... - PESHAWAR...]

`SELECT * FROM student ORDER BY city`

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: | Sort by key: None

+ Options

					Id	Firstname	Lastname	Email	DateOfBirth	reg_date	City
<input type="checkbox"/>					3	Fahad	Iqbal	fahad@gmail.com	1991-03-15	2021-03-25 14:45:36	Karachi
<input type="checkbox"/>					1	Afaq	Shah	afaq@gmail.com	1998-03-12	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					4	Zahid	Irfan	zahid@gmail.com	1995-05-23	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					6	Fahad	Khan	fahad@gmail.com	1994-01-28	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					2	Asad	Khan	asad@gmail.com	1998-03-12	2021-03-25 14:45:36	Peshawar
<input type="checkbox"/>					5	Saad	Akhtar	saad@gmail.com	1994-01-28	2021-03-25 14:45:36	Peshawar

Example 2 (Order by DESC):

✓ Showing rows 0 - 5 (6 total, Query took 0.0018 seconds.) [City: PESHAWAR... - KARACHI...]

`SELECT * FROM student ORDER BY city DESC`

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: | Sort by key: None

+ Options

					Id	Firstname	Lastname	Email	DateOfBirth	reg_date	City
<input type="checkbox"/>					2	Asad	Khan	asad@gmail.com	1998-03-12	2021-03-25 14:45:36	Peshawar
<input type="checkbox"/>					5	Saad	Akhtar	saad@gmail.com	1994-01-28	2021-03-25 14:45:36	Peshawar
<input type="checkbox"/>					1	Afaq	Shah	afaq@gmail.com	1998-03-12	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					4	Zahid	Irfan	zahid@gmail.com	1995-05-23	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					6	Fahad	Khan	fahad@gmail.com	1994-01-28	2021-03-25 14:45:36	Lahore
<input type="checkbox"/>					3	Fahad	Iqbal	fahad@gmail.com	1991-03-15	2021-03-25 14:45:36	Karachi

THE END