What is SQL?

* SQL stands for Structured Query Language
* SQL lets you access and manipulate databases

What Can SQL do?

* SQL can execute queries against a database
* SQL can retrieve data from a database
* SQL can insert records in a database
* SQL can update records in a database
* SQL can delete records from a database
* SQL can create new databases
* SQL can create new tables in a database
* SQL can create stored procedures in a database
* SQL can create views in a database
* SQL can set permissions on tables, procedures, and views

commands (such as SELECT, UPDATE, DELETE, INSERT, WHERE)

* SELECT - extracts data from a database
* UPDATE - updates data in a database
* DELETE - deletes data from a database
* INSERT INTO - inserts new data into a database
* CREATE DATABASE - creates a new database
* ALTER DATABASE - modifies a database
* CREATE TABLE - creates a new table
* ALTER TABLE - modifies a table
* DROP TABLE - deletes a table
* CREATE INDEX - creates an index (search key)
* DROP INDEX - deletes an index

DISTINCT:-> Select all the different from table ( not allow duplicates)

Ex. SELECT DISTINCT Country FROM Customers;

The WHERE clause is used to filter records.(where define condtion/resource)

|  |  |  |
| --- | --- | --- |
| **Operator** | **Description** | **Example** |
| = | Equal | [Try it](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_equal_to) |
| > | Greater than | [Try it](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_greater_than) |
| < | Less than | [Try it](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_less_than) |
| >= | Greater than or equal | [Try it](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_greater_than2) |
| <= | Less than or equal | [Try it](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_less_than2) |
| <> | Not equal. **Note:** In some versions of SQL this operator may be written as != | [Try it](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_not_equal_to) |
| BETWEEN | Between a certain range | [Try it](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_between) |
| LIKE | Search for a pattern | [Try it](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_like) |
| IN | To specify multiple possible values for a column |  |

Ex. SELECT \* FROM Customers

WHERE City LIKE 's%'; 🡪 it well select all data from city name start with “S”

The LIKE operator is used in a WHERE clause to search for a specified pattern in a column.

There are two wildcards often used in conjunction with the LIKE operator:

* The percent sign % represents zero, one, or multiple characters
* The underscore sign \_ represents one, single character

The ORDER BY keyword is used to sort the result-set in ascending or descending order.(ASC|DESC)

The WHERE clause can contain one or many AND operators.

The AND operator is used to filter records based on more than one condition, like if you want to return all customers from Spain that starts with the letter 'G':

The WHERE clause can contain one or more OR operators.

The OR operator is used to filter records based on more than one condition, like if you want to return all customers from Germany but also those from Spain:

Ex.

SELECT \*FROM Customers  
WHERE Country = 'Germany'  AND/OR Country = 'Spain';

The NOT operator is used in combination with other operators to give the opposite result, also called the negative result.

Ex. ,we want to return all customers that are NOT from Spain:

INSERT INTO

1. Specify both the column names and the values to be inserted:

INSERT INTO *table\_name* (*column1*,*column2*,*column3*, ...)  
VALUES (*value1*,*value2*,*value3*, ...);

2. If you are adding values for all the columns of the table, you do not need to specify the column names in the SQL query. However, make sure the order of the values is in the same order as the columns in the table. Here, the INSERT INTO syntax would be as follows:

INSERT INTO *table\_name*  
VALUES (*value1*,*value2*,*value3*, ...);

IS NULL Syntax

SELECT *column\_names*FROM *table\_name*  
WHERE *column\_name* IS NULL;

IS NOT NULL Syntax

SELECT *column\_names*FROM *table\_name*  
WHERE *column\_name* IS NOT NULL;

Update table:

UPDATE Customers  
SET ContactName = 'Alfred Schmidt', City= 'Frankfurt'  
WHERE CustomerID = 1;

DELETE FROM Customers WHERE CustomerName='Alfreds Futterkiste';

SELECT TOP 50 PERCENT \* FROM Customers;

SQL Aggregate Functions

An aggregate function is a function that performs a calculation on a set of values, and returns a single value.

The most commonly used SQL aggregate functions are:

* MIN() - returns the smallest value within the selected column
* MAX() - returns the largest value within the selected column
* COUNT() - returns the number of rows in a set
* SUM() - returns the total sum of a numerical column
* AVG() - returns the average value of a numerical column

Aggregate functions ignore null values (except for COUNT()).

The MIN() function returns the smallest value of the selected column.

The MAX() function returns the largest value of the selected column.

The COUNT() function returns the number of rows that matches a specified criterion.

The SUM() function returns the total sum of a numeric column.

The AVG() function returns the average value of a numeric column.

The BETWEEN operator selects values within a given range. The values can be numbers, text, or dates.

The BETWEEN operator is inclusive: begin and end values are included.

SQL Aliases

SQL aliases are used to give a table, or a column in a table, a temporary name.

Aliases are often used to make column names more readable.

An alias only exists for the duration of that query.

An alias is created with the AS keyword.

SQL statement (that contains an INNER JOIN), that selects records that have matching values in both tables:

Ex: SELECT Orders.OrderID, Customers.CustomerName, Orders.OrderDate  
FROM Orders  
INNER JOIN Customers ON Orders.CustomerID=Customers.CustomerID;



In some databases LEFT JOIN is called LEFT OUTER JOIN.



**note:** In some databases RIGHT JOIN is called RIGHT OUTER JOIN.





A self join is a regular join, but the table is joined with itself.

Self Join Syntax

SELECT *column\_name(s)*  
FROM *table1 T1, table1 T2*  
WHERE *condition*;

**SQL DATABASE:**

The SQL CREATE DATABASE Statement

The CREATE DATABASE statement is used to create a new SQL database.

Syntax

CREATE DATABASE *databasename***;**

The SQL DROP DATABASE Statement

The DROP DATABASE statement is used to drop an existing SQL database.

Syntax

DROP DATABASE *databasename***;**

The SQL BACKUP DATABASE Statement

The BACKUP DATABASE statement is used in SQL Server to create a full back up of an existing SQL database.

Syntax

BACKUP DATABASE *databasename*  
TO DISK = '*filepath*';

SQL PRIMARY KEY Constraint

The PRIMARY KEY constraint uniquely identifies each record in a table.

Primary keys must contain UNIQUE values, and cannot contain NULL values.

A table can have only ONE primary key; and in the table, this primary key can consist of single or multiple columns (fields).

SQL FOREIGN KEY Constraint

The FOREIGN KEY constraint is used to prevent actions that would destroy links between tables.

A FOREIGN KEY is a field (or collection of fields) in one table, that refers to the [PRIMARY KEY](https://www.w3schools.com/sql/sql_primarykey.asp) in another table.

The table with the foreign key is called the child table, and the table with the primary key is called the referenced or parent table

CREATE TABLE Orders (  
    OrderID int NOT NULL,  
    OrderNumber int NOT NULL,  
    PersonID int,  
    PRIMARY KEY (OrderID),  
    FOREIGN KEY (PersonID) REFERENCES Persons(PersonID)  
);

Q/A

Question 1:

What does SQL stand for?

Structured Query Language    Your answer

Strong Question Language

Structured Question Language

Question 2:

Which SQL statement is used to extract data from a database?

SELECT    Your answer

OPEN

EXTRACT

GET

Question 3:

Which SQL statement is used to update data in a database?

UPDATE    Your answer

SAVE

MODIFY

SAVE AS

Question 4:

Which SQL statement is used to delete data from a database?

DELETE    Your answer

COLLAPSE

REMOVE

Question 5:

Which SQL statement is used to insert new data in a database?

INSERT INTO    Your answer

ADD RECORD

ADD NEW

INSERT NEW

Question 6:

With SQL, how do you select a column named "FirstName" from a table named "Persons"?

SELECT FirstName FROM Persons    Your answer

EXTRACT FirstName FROM Persons

SELECT Persons.FirstName

Question 7:

With SQL, how do you select all the columns from a table named "Persons"?

SELECT \* FROM Persons    Your answer

SELECT Persons

SELECT [all] FROM Persons

SELECT \*.Persons

Question 8:

With SQL, how do you select all the records from a table named "Persons" where the value of the column "FirstName" is "Peter"?

SELECT \* FROM Persons WHERE FirstName='Peter'    Your answer

SELECT [all] FROM Persons WHERE FirstName LIKE 'Peter'

SELECT [all] FROM Persons WHERE FirstName='Peter'

SELECT \* FROM Persons WHERE FirstName<>'Peter'

Question 9:

With SQL, how do you select all the records from a table named "Persons" where the value of the column "FirstName" starts with an "a"?

SELECT \* FROM Persons WHERE FirstName LIKE '%a'    Your answer

SELECT \* FROM Persons WHERE FirstName='a'

SELECT \* FROM Persons WHERE FirstName='%a%'

SELECT \* FROM Persons WHERE FirstName LIKE 'a%'    Correct answer

Question 10:

The OR operator displays a record if ANY conditions listed are true. The AND operator displays a record if ALL of the conditions listed are true

True    Your answer

False

Question 11:

With SQL, how do you select all the records from a table named "Persons" where the "FirstName" is "Peter" and the "LastName" is "Jackson"?

SELECT \* FROM Persons WHERE FirstName='Peter' AND LastName='Jackson'    Your answer

SELECT \* FROM Persons WHERE FirstName<>'Peter' AND LastName<>'Jackson'

SELECT FirstName='Peter', LastName='Jackson' FROM Persons

Question 12:

With SQL, how do you select all the records from a table named "Persons" where the "LastName" is alphabetically between (and including) "Hansen" and "Pettersen"?

SELECT \* FROM Persons WHERE LastName BETWEEN 'Hansen' AND 'Pettersen'    Your answer

SELECT \* FROM Persons WHERE LastName>'Hansen' AND LastName<'Pettersen'

SELECT LastName>'Hansen' AND LastName<'Pettersen' FROM Persons

Question 13:

Which SQL statement is used to return only different values?

SELECT DISTINCT    Your answer

SELECT DIFFERENT

SELECT UNIQUE

Question 14:

Which SQL keyword is used to sort the result-set?

ORDER BY    Your answer

ORDER

SORT BY

SORT

Question 15:

With SQL, how can you return all the records from a table named "Persons" sorted descending by "FirstName"?

SELECT \* FROM Persons ORDER BY FirstName DESC    Your answer

SELECT \* FROM Persons SORT 'FirstName' DESC

SELECT \* FROM Persons ORDER FirstName DESC

SELECT \* FROM Persons SORT BY 'FirstName' DESC

Question 16:

With SQL, how can you insert a new record into the "Persons" table?

INSERT INTO Persons VALUES ('Jimmy', 'Jackson')    Your answer

INSERT ('Jimmy', 'Jackson') INTO Persons

INSERT VALUES ('Jimmy', 'Jackson') INTO Persons

Question 17:

With SQL, how can you insert "Olsen" as the "LastName" in the "Persons" table?

INSERT INTO Persons (LastName) VALUES ('Olsen')    Your answer

INSERT INTO Persons ('Olsen') INTO LastName

INSERT ('Olsen') INTO Persons (LastName)

Question 18:

How can you change "Hansen" into "Nilsen" in the "LastName" column in the Persons table?

UPDATE Persons SET LastName='Nilsen' WHERE LastName='Hansen'    Your answer

MODIFY Persons SET LastName='Hansen' INTO LastName='Nilsen

MODIFY Persons SET LastName='Nilsen' WHERE LastName='Hansen'

UPDATE Persons SET LastName='Hansen' INTO LastName='Nilsen'

Question 19:

With SQL, how can you delete the records where the "FirstName" is "Peter" in the Persons Table?

DELETE FROM Persons WHERE FirstName = 'Peter'    Your answer

DELETE FirstName='Peter' FROM Persons

DELETE ROW FirstName='Peter' FROM Persons

Question 20:

With SQL, how can you return the number of records in the "Persons" table?

SELECT COUNT(\*) FROM Persons    Your answer

SELECT NO(\*) FROM Persons

SELECT LEN(\*) FROM Persons

SELECT COLUMNS(\*) FROM Persons

Question 21:

What is the most common type of join?

INNER JOIN    Your answer

JOINED

JOINED TABLE

INSIDE JOIN

Question 22:

Which operator is used to select values within a range?

BETWEEN    Your answer

RANGE

WITHIN

Question 23:

The NOT NULL constraint enforces a column to not accept NULL values.

True    Your answer

False

Question 24:

Which operator is used to search for a specified pattern in a column?

LIKE    Your answer

FROM

GET

Question 25:

Which SQL statement is used to create a database table called 'Customers'?

CREATE DATABASE TABLE Customers    Your answer

CREATE DB Customers

CREATE TABLE Customers    Correct answer

CREATE DATABASE TAB Customers