1. What is Database?

Ans: A database is a collection of [information](http://searchsqlserver.techtarget.com/definition/information) that is organized so that it can easily be accessed, managed, and updated. In one view, databases can be classified according to types of content.

1. What is Table?

Ans: A [table](https://en.wikipedia.org/wiki/Table_%28information%29) is a collection of related data held in a structured format within a database. It consists of [columns](https://en.wikipedia.org/wiki/Column_%28database%29), and [rows](https://en.wikipedia.org/wiki/Row_%28database%29).

Table" is another term for ["relation"](https://en.wikipedia.org/wiki/Relation_%28database%29); although there is the difference in that a table is usually a [multiset](https://en.wikipedia.org/wiki/Multiset) (bag) of rows where a relation is a [set](https://en.wikipedia.org/wiki/Set_%28computer_science%29) and does not allow duplicates. Besides the actual data rows, tables generally have associated with them some [metadata](https://en.wikipedia.org/wiki/Metadata), such as [constraints](https://en.wikipedia.org/wiki/Check_constraint) on the table or on the values within particular columns.

The data in a table does not have to be physically stored in the database. [Views](https://en.wikipedia.org/wiki/View_%28database%29) also function as relational tables, but their data are calculated at query time.

1. What is column?

Ans: A **column** is a set of [data](https://en.wikipedia.org/wiki/Data) values of a particular simple [type](https://en.wikipedia.org/wiki/Datatype), one for each [row](https://en.wikipedia.org/wiki/Row_%28database%29) of the table. The columns provide the structure according to which the rows are composed. When a column allows data values of a single type, it does not essentially mean it only has simple text values.

1. What is Row?

Ans: Row is a horizontal group of values within a table. It contains values for multiple fields, which are defined by columns. Because rows contain data from multiple columns, in databases, each table row may be considered a record. For example, a row (or record) from a Employee table may contain an employee's name, address, position, salary, and other information. When [querying](https://techterms.com/definition/query) a database, the results are typically returned as an array of rows, which is similar to a group of records. Individual values can be accessed by selecting a specific column (or field) within a row.

1. Example for Innerjoin?

Ans: SELECT column\_name(s)  
FROM table1  
INNER JOIN table2  
ON table1.column\_name=table2.column\_name;

Example:

SELECT Customers.CustomerName, Orders.OrderID  
FROM Customers  
INNER JOIN Orders  
ON Customers.CustomerID=Orders.CustomerID  
ORDER BY Customers.CustomerName;

6) Example for Left Outer Join

Ans: SELECT column\_name(s)  
FROM table1  
LEFT OUTER JOIN table2  
ON table1.column\_name=table2.column\_name;

Exaample:

SELECT Customers.CustomerName, Orders.OrderID  
FROM Customers  
LEFT JOIN Orders  
ON Customers.CustomerID=Orders.CustomerID  
ORDER BY Customers.CustomerName;

7) Example for Right Outer Join

Ans: *SELECT* column\_name(s) *FROM* table1 *RIGHT OUTER JOIN* table2 *ON* table1.column\_name*=*table2.column\_name*;*

Example: SELECT Orders.OrderID, Employees.FirstName  
FROM Orders  
RIGHT JOIN Employees  
ON Orders.EmployeeID=Employees.EmployeeID  
ORDER BY Orders.OrderID;

8) Eamples for sum,avg,maximum

Syntax Sum():

SELECT SUM(column\_name) FROM table\_name;

Ex: SELECT SUM(Quantity) AS TotalItemsOrdered FROM OrderDetails;

Avg() Syntax:

SELECT AVG(column\_name) FROM table\_name

Ex: SELECT AVG(Price) AS PriceAverage FROM Products;

Max() Syntax:

SELECT MAX(column\_name) FROM table\_name;

Ex: SELECT MAX(Price) AS HighestPrice FROM Products;

9) Example for GroupBy

Syntax:

SELECT column\_name, aggregate\_function(column\_name)  
FROM table\_name  
WHERE column\_name operator value  
GROUP BY column\_name;

Ex: SELECT Shippers.ShipperName,COUNT(Orders.OrderID) AS NumberOfOrders FROM Orders  
LEFT JOIN Shippers  
ON Orders.ShipperID=Shippers.ShipperID  
GROUP BY ShipperName;

10) Example for Having

Syntax:

SELECT column\_name, aggregate\_function(column\_name)  
FROM table\_name  
WHERE column\_name operator value  
GROUP BY column\_name  
HAVING aggregate\_function(column\_name) operator value;

Ex: SELECT Employees.LastName, COUNT(Orders.OrderID) AS NumberOfOrders FROM (Orders  
INNER JOIN Employees  
ON Orders.EmployeeID=Employees.EmployeeID)  
GROUP BY LastName  
HAVING COUNT(Orders.OrderID) > 10;

11) Example for where Condition

Syntax:

SELECT column\_name,column\_name  
FROM table\_name  
WHERE column\_name operator value;

Ex:

SELECT \* FROM Customers  
WHERE Country='Mexico';

12) Example for PRIMARY KEY

Ans:

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

Primary keys must contain UNIQUE values.

A primary key column cannot contain NULL values.

Most tables should have a primary key, and each table can have only ONE primary key.

## SQL PRIMARY KEY Constraint on CREATE TABLE

The following SQL creates a PRIMARY KEY on the "P\_Id" column when the "Persons" table is created:

**MySQL:**

CREATE TABLE Persons  
(  
P\_Id int NOT NULL,  
LastName varchar(255) NOT NULL,  
FirstName varchar(255),  
Address varchar(255),  
City varchar(255),  
PRIMARY KEY (P\_Id)  
)

**SQL Server / Oracle / MS Access:**

CREATE TABLE Persons  
(  
P\_Id int NOT NULL PRIMARY KEY,  
LastName varchar(255) NOT NULL,  
FirstName varchar(255),  
Address varchar(255),  
City varchar(255)  
)

To allow naming of a PRIMARY KEY constraint, and for defining a PRIMARY KEY constraint on multiple columns, use the following SQL syntax:

**MySQL / SQL Server / Oracle / MS Access:**

CREATE TABLE Persons  
(  
P\_Id int NOT NULL,  
LastName varchar(255) NOT NULL,  
FirstName varchar(255),  
Address varchar(255),  
City varchar(255),  
CONSTRAINT pk\_PersonID PRIMARY KEY (P\_Id,LastName)  
)

**Note:** In the example above there is only ONE PRIMARY KEY (pk\_PersonID). However, the VALUE of the primary key is made up of TWO COLUMNS (P\_Id + LastName).

## SQL PRIMARY KEY Constraint on ALTER TABLE

To create a PRIMARY KEY constraint on the "P\_Id" column when the table is already created, use the following SQL:

**MySQL / SQL Server / Oracle / MS Access:**

ALTER TABLE Persons  
ADD PRIMARY KEY (P\_Id)

To allow naming of a PRIMARY KEY constraint, and for defining a PRIMARY KEY constraint on multiple columns, use the following SQL syntax:

**MySQL / SQL Server / Oracle / MS Access:**

ALTER TABLE Persons  
ADD CONSTRAINT pk\_PersonID PRIMARY KEY (P\_Id,LastName)

**Note:** If you use the ALTER TABLE statement to add a primary key, the primary key column(s) must already have been declared to not contain NULL values (when the table was first created).

## To DROP a PRIMARY KEY Constraint

To drop a PRIMARY KEY constraint, use the following SQL:

**MySQL:**

ALTER TABLE Persons  
DROP PRIMARY KEY

**SQL Server / Oracle / MS Access:**

ALTER TABLE Persons  
DROP CONSTRAINT pk\_PersonID

13) Example for FOREIGN KEY

Ans: The following SQL creates a FOREIGN KEY on the "P\_Id" column when the "Orders" table is created:

**MySQL:**

CREATE TABLE Orders  
(  
O\_Id int NOT NULL,  
OrderNo int NOT NULL,  
P\_Id int,  
PRIMARY KEY (O\_Id),  
FOREIGN KEY (P\_Id) REFERENCES Persons(P\_Id)  
)

**SQL Server / Oracle / MS Access:**

CREATE TABLE Orders  
(  
O\_Id int NOT NULL PRIMARY KEY,  
OrderNo int NOT NULL,  
P\_Id int FOREIGN KEY REFERENCES Persons(P\_Id)  
)

To allow naming of a FOREIGN KEY constraint, and for defining a FOREIGN KEY constraint on multiple columns, use the following SQL syntax:

**MySQL / SQL Server / Oracle / MS Access:**

CREATE TABLE Orders  
(  
O\_Id int NOT NULL,  
OrderNo int NOT NULL,  
P\_Id int,  
PRIMARY KEY (O\_Id),  
CONSTRAINT fk\_PerOrders FOREIGN KEY (P\_Id)  
REFERENCES Persons(P\_Id)  
)

## SQL FOREIGN KEY Constraint on ALTER TABLE

To create a FOREIGN KEY constraint on the "P\_Id" column when the "Orders" table is already created, use the following SQL:

**MySQL / SQL Server / Oracle / MS Access:**

ALTER TABLE Orders  
ADD FOREIGN KEY (P\_Id)  
REFERENCES Persons(P\_Id)

To allow naming of a FOREIGN KEY constraint, and for defining a FOREIGN KEY constraint on multiple columns, use the following SQL syntax:

**MySQL / SQL Server / Oracle / MS Access:**

ALTER TABLE Orders  
ADD CONSTRAINT fk\_PerOrders  
FOREIGN KEY (P\_Id)  
REFERENCES Persons(P\_Id)

## To DROP a FOREIGN KEY Constraint

To drop a FOREIGN KEY constraint, use the following SQL:

**MySQL:**

ALTER TABLE Orders  
DROP FOREIGN KEY fk\_PerOrders

**SQL Server / Oracle / MS Access:**

ALTER TABLE Orders  
DROP CONSTRAINT fk\_PerOrders

14) Finding second highest salary from row table

Ans:

SELECT TOP 1 salary FROM ( SELECT TOP 2 salary FROM employees ORDER BY salary DESC) AS emp ORDER BY salary ASC