1. What is SQL?

SQL stand Structure Query Language. SQl is Standard language for accessing and manipulating databases.

1. What is Database?

A Database is an organized collection of data,stored and retrived digitally from a remote or local computer system.

1. What is Where Clause?

The SQL **WHERE** clause is used to specify a condition while fetching the data from a single table or by joining with multiple tables. You should use the WHERE clause to filter the records and fetching only the necessary records. The WHERE clause is not only used in the SELECT statement, but it is also used in the UPDATE, DELETE statement.

Syntax,

Select ID, Name, Salary from Customer where Name=’Test’;

1. SQL And, Or and Not Clause

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

Syntax,

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

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

Syntax,

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

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

Syntax,

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

* Combine AND,NOT,OR

Syntax,

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

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

1. Select Top Clause

The SELECT TOP clause is used to specify the number of records to return.

Syntax,

SELECT TOP 50 PERCENT \* FROM Customers;

1. SQL Min() and Max()

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

Syntax,

SELECT MIN(Price) AS SmallestPrice  
FROM Products;

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

Syntax,

SELECT MAX(Price) AS LargestPrice  
FROM Products;

1. Aliases

* You can rename a table or a column temporarily by giving another name known as **Alias**. The use of table aliases is to rename a table in a specific SQL statement. The renaming is a temporary change and the actual table name does not change in the database. The column aliases are used to rename a table's columns for the purpose of a particular SQL query.
* An alias is created with the AS keyword.

Syntax,

SELECT o.OrderID, o.OrderDate, c.CustomerName  
FROM Customers AS c, Orders AS o  
WHERE c.CustomerName='Around the Horn' AND c.CustomerID=o.CustomerID;

1. Joins

* A JOIN clause is used to combine rows from two or more tables, based on a related column between them.
* 4 Types of Joins:
* 1. Inner Join :
* The INNER JOIN keyword selects records that have matching values in both tables.

Syntax,

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

* 2. Left Join:
* The LEFT JOIN keyword returns all records from the left table (table1), and the matching records from the right table (table2). The result is 0 records from the right side, if there is no match.

Syntax,

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

* 3.Right Join:
* The RIGHT JOIN keyword returns all records from the right table (table2), and the matching records from the left table (table1). The result is 0 records from the left side, if there is no match.

Syntax,

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

* 4.Full(Outer) Join:
* The FULL OUTER JOIN keyword returns all records when there is a match in left (table1) or right (table2) table records.

Syntax,

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

1. Union

* The SQL UNION clause/operator is used to combine the results of two or more SELECT statements without returning any duplicate rows.
* Every SELECT statement within UNION must have the same number of columns
* The columns must also have similar data types
* The columns in every SELECT statement must also be in the same order

Syntax,

SELECT City FROM Customers  
UNION  
SELECT City FROM Suppliers  
ORDER BY City;

1. Intersect

* In simple words, the INTERSECT statement will return only those rows which will be common to both of the SELECT statements.

Syntax,

**SELECT** name **FROM** Students /\* Fetch names from students \*/

**INTERSECT** /\* that are present in contacts as well \*/

**SELECT** name **FROM** Contacts;

1. Group By

* The GROUP BY statement is often used with aggregate functions (COUNT(), MAX(), MIN(), SUM(), AVG()) to group the result-set by one or more columns.

Syntax,

SELECT COUNT(CustomerID), Country  
FROM Customers  
GROUP BY Country  
ORDER BY COUNT(CustomerID) DESC;

1. Having

* clause in SQL is used to filter records in combination with the GROUP BY clause. It is different from WHERE, since the WHERE clause cannot filter aggregated records.

Syntax,

**SELECT** COUNT(studentId), country

**FROM** myDB.students

**WHERE** country != "IN**D**IA"

**GROUP** **BY** country

**HAVING** COUNT(studentID) > 5;

1. Operators

* Arithmetic Operator
* Bitwise Operator
* Comparison Operator
* Compound Operator
* Logical Operator

1. Index

* The Onely gold of index to make a search faster.
* Indexes are used to retrieve data from the database more quickly than otherwise. The users cannot see the indexes, they are just used to speed up searches/queries.

Syntax,

CREATE INDEX idx\_lastname  
ON Persons (LastName);

* Two Types of Index:

1.Cluster Index

2.Non-Cluster Index

1. What is Tables and Fields?

A table is an organized collection of data stored in the form of rows and columns. Columns can be categorized as vertical and rows as horizontal. The columns in a table are called fields while the rows can be referred to as records.

1. What are the subsets of SQL?

The main significant subsets of SQL are:

1. **DDL**(Data Definition Language)
2. **DML**(Data Manipulation Language)
3. **DCL**(Data Control Language)
4. **TCL**(Transaction Control Language)
5. Access Modifiers

  Access Modifiers is used to set the access level/visibility for classes, fields, methods and properties.

|  |  |
| --- | --- |
| **Modifier** | **Description** |
| public | The code is accessible for all classes |
| private | The code is only accessible within the same class |
| protected | The code is accessible within the same class, or in a class that is inherited from that class. |
| internal | The code is only accessible within its own assembly, but not from another assembly. |

1. Oops Concept

object-oriented programming is about creating objects that contain both data and methods.

Object-oriented programming has several advantages over procedural programming:

* OOP is faster and easier to execute
* OOP provides a clear structure for the programs
* OOP helps to keep the C# code DRY "Don't Repeat Yourself", and makes the code easier to maintain, modify and debug
* OOP makes it possible to create full reusable applications with less code and shorter development time
  1. Inheritance
     + It is possible to inherit fields and methods from one class to another.
  2. Polymorphism
     + Polymorphism means "many forms", and it occurs when we have many classes that are related to each other by inheritance.
     + [**Inheritance**](https://www.w3schools.com/cs/cs_inheritance.asp) lets us inherit fields and methods from another class. **Polymorphism** uses those methods to perform different tasks. This allows us to perform a single action in different ways.
     + Two Types of Polymorphism:
       - 1. Run-time Polymorphism:
         * Different method and Different parameter known as method overloading (Run-time Polymorphism).
       - 2. Compile-time Polymorphism :
         * In c#, Compile Time Polymorphism means **defining multiple methods with the same name but with different parameters**. Using compile-time polymorphism, we can perform different tasks with the same method name by passing different parameters.(Method Overridding).
  3. Abstraction
     + Abstraction in C# is **the process to hide the internal details and showing only the functionality.**
     + Ex. Phone calls
  4. Ensapsulation
     + Encapsulation is the concept of wrapping data into a single unit. It collects data members and member functions into a single unit called class. The purpose of encapsulation is to prevent alteration of data from outside. This data can only be accessed by getter functions of the class.
     + A fully encapsulated class has getter and setter functions that are used to read and write data. This class does not allow data access directly.
     + Real world Ex. Ecapsual and Class

1. Self Join

* The SQL Self join is used to join a table to itself as if the table were two tables;temporaoily renaming at least one table in the SQL Statement.

Syntax,

SELECT A.CustomerName AS CustomerName1, B.CustomerName AS CustomerName2, A.City  
FROM Customers A, Customers B  
WHERE A.CustomerID <> B.CustomerID  
AND A.City = B.City  
ORDER BY A.City;

1. Cross Join

* In SQL, the **CROSS JOIN** is used to combine each row of the first table with each row of the second table. It is also known as the Cartesian join since it returns the Cartesian product of the sets of rows from the joined tables.

Syntax,

SELECT Customers.CustomerName, Orders.OrderID

FROM Customers

CROSS JOIN Orders

WHERE Customers.CustomerID = Orders.CustomerID;

22. Select distinct

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

Syntax,

SELECT distinct Country FROM Customers;

23. Constructor

* A method that is called when an instance of a class is created.  Usually we put the initialization code in the constructor. The name of the constructor is always is the same name as the class. A C# constructor can be public or private. A class can have multiple overloaded constructors.
* Ex.

**public** **class** mySampleClass

   {

**public** mySampleClass()

      {

         // This is the constructor method.

      }

   // rest of the class members goes here.

}

24. Properties

Property in C# is a member of a class that provides a flexible mechanism for classes to expose private fields. Internally, C# properties are special methods called accessors. A C# property have two accessors, get property accessor and set property accessor. A get accessor returns a property value, and a set accessor assigns a new value.

Eg.

class Person

{

private string name; // field

public string Name // property

{

get { return name; } // get method

set { name = value; } // set method

}

}

### 25. What is enum in C#?

* An enum is a value type with a set of related named constants often referred to as an enumerator list. The enum keyword is used to declare an enumeration. It is a primitive data type that is user-defined.
* An enum type can be an integer (float, int, byte, double, etc.). But if you use it beside int it has to be cast.
* An enum is used to create numeric constants in the .NET framework. All the members of enum are enum type. There must be a numeric value for each enum type.
* The default underlying type of the enumeration element is int. By default, the first enumerator has the value 0, and the value of each successive enumerator is increased by 1.

26. Interface? And why do we use interface?

* + Interface is an abstract and a public class in which methods are declared
  + A class implementing the interface defines all the methods declared in an interface
  + The major reasons to use an interface is to:
    1. Make secure applications- because with interface only required information is exposed to the user
    2. Multiple inheritance- as we know multiple inheritance is not supported in c#, interfaces can help achieve that functionality as one class can inherent multiple interface

27. Aggregate Function

An aggregate function performs a calculation one or more values and returns a single value. The aggregate function is often used with the [GROUP BY](https://www.sqlservertutorial.net/sql-server-basics/sql-server-group-by/) clause and [HAVING](https://www.sqlservertutorial.net/sql-server-basics/sql-server-having/) clause of the [SELECT](https://www.sqlservertutorial.net/sql-server-basics/sql-server-select/) statement.

28. What is Asp.Net Core Defination

ASP.NET Core is the open-source version of ASP.NET, that runs on macOS, Linux, and Windows. ASP.NET Core was first released in 2016 and is a re-design of earlier Windows-only versions of ASP.NET.

29. Asp.Net Core Advantages/Feature

.NET Core is Cross-Platform .NET Core is Open Source Mature Framework and Widely Used Programming Languages The .NET Core Supports a Wide Range of Application Types Increased Security with .NET Core .NET Core Enables Top App Performance .NET Core Enables Flexibility .NET Core is Cost Effective .NET Core has a Large Community .NET Core was Created by Microsoft

30. What is Query?

A query is **a request for data or information from a database table or combination of tables**. This data may be generated as results returned by Structured Query Language (SQL)

31. Cursor

* Cursor is a Temporary Memory or Temporary Work Station.
* A SQL cursor is a database object that is used to retrieve data from a result set one row at a time.
* Why Used??
* A SQL cursor is used when the data needs to be updated row by row.
* Cursors are used to store Database Tables.
* It allows you to process individual row returned by a query.
* It is Allocated by Database server at the Time of performing DML operations on Table by User.
* Two types of cursor:

1. Implicit Cursor

* Implicit Cursors are also known as Default Cursors of SQL Server.These Cursors are allocated by SQL SERVER when the user performs DML operations.

1. Explicit Cursor

* Explicit Cursor are created by users whenever the user requires them.Explicit Cursors are used for Fetching data from Table in Row-By-Row Manner.
* Methods of Cursor :

1. Next
2. Prior
3. First
4. Last
5. Absolute n
6. Relative n

* SQL cursor life cycle :

1. Declaring Cursor

* A cursor is declared by defining the SQL statement.

1. Opening Cursor

* A cursor is opened for storing data retrieved from the result set.

1. Fetching Cursor

* When a cursor is opened, rows can be fetched from the cursor one by one or in a block to do data manipulation.

1. Closing Cursor

* The cursor should be closed explicitly after data manipulation.

1. De-allocating Cursor

* Cursors should be deallocated to delete cursor definition and relese all the system resources associated with the cursor.
* Ex.

declar mycursor cursor scroll for select \* from EmployeeTbl

open mycursor

fetch first from mycursor

fetch next from mycursor

fetch piror from mycursor

fetch Last from mycursor

fetch Absolute 4 from mycursor

fetch Relative 3 from mycursor

close mycursor

deallocated mycursor

33. SQL View

* A view is nothing more than a saved SQL query. A view can also be considered as a virtual table

Ex.

Create view VWEmployeesByDepartment

As

Select Id,Name,Salary,Gender,DeptName

From tblEmployee

Join tblDepartment

On tblEmployee.DepartmentId = tblDepartment.DepartmentId

* Advantages of views

1. Views can be used to reduce the complexity of the database schema
2. Views can be used as a mechanism to implement row and column level security
3. Views can be used to present aggregated data and hide details data.

34. Normalization

* Normalization is the process of organizing data in the database.
* Why used ?
* Minimize redundancy in relation.
* Divide big table into small table.
* Types:

1NF

2NF

3NF

SQL Injections Video

Dependency Injections Video