

C#

Q. What is difference between web Farm and web garden?

A. For big sites where there are millions of daily users we need to host the sites on multiple servers. This is called **Web Farms**.

All IIS Request are processed by worker process (w3wp.exe). By default each and every application pool contains single worker process. But an application pool with multiple worker process is called **Web Garden**.

Q. What is Abstract or Must Inherit Class?

A. An abstract class is the one that is not used to create objects. An abstract class is designed to act as a base class (to be inherited by other classes). Abstract classes are similar to interfaces. After declaring an abstract class, it cannot be instantiated on it's own, it must be inherited

Q. What are Static functions?

A. A static function is one that can be called directly using class name; without creating its object.

Q. What is Static or Shared Class?

A. A static class is one in which all data members and member functions are static. We can call the functions of this class directly without creating its object.

Q. What is an Interface?

A. Interfaces help to define the various properties, methods and events that classes are able to implement. For developers these fine a small group of closely related properties, methods, and events. Additional feature and functionality can be added at any by adding additional interfaces and implementations.

Q. What is sealed class?

A. Sealed class is a class that we cannot inherit. A class, which restricts inheritance for security reason is declared, sealed class. A sealed class cannot be an abstract class.

Q. What is a constructor?

A. It is a method in the class which gets executed automatically when the object of class is created. Usually we put the initialization code in the constructor. Constructor name is same as the Class Name.

Q. What is the need of declaring a method with keyword virtual?

A. The method or property can be overridden.

Q. What are Parameterized Constructors?

A. Parameterized constructors (or more simply "constructors") allow you to create a new instance of a class while simultaneously passing arguments to the new instance.

Q. Can Static class have Constructor?

A. Yes, Static class can have only static constructors. A static constructor is used to initialize any static data, or to perform a particular action that needs to be performed once only. It is called automatically before the first instance is created or any static members are referenced.

Q. What is Structure?

A. A struct is a simple user-defined type. Structures are value Type. It's a lightweight alternative to a class. They may implement an Interface. They can be instantiated without using a new operator.

Q. What is the difference between Structure and Class?

A. Structs are value types and classes are reference types.

1. The general different is that a reference type lives on the heap, and a value type lives inline (i.e. wherever your variable or field is defined).
2. A variable containing a value type contains the entire value type value. A variable containing a reference type contains a pointer or a reference to somewhere else in memory where the actual value resides.

Q. Can structures have default constructor?

A. No

Q. What is function overloading?

A. C# allows us to define multiple functions with the same name differing in the number type and order of arguments. This is termed as function overloading.

Q. What is Function Overriding?

A. Method overriding in C# is a feature like the virtual function in C++. Method overriding is a feature that allows you to invoke functions (that have the same signatures) that belong to different classes in the same hierarchy of inheritance using the base class reference. C# makes use of two keywords: virtual and overrides to accomplish Method overriding.

Q. What is Virtual Functions?

A. If we want override the definition of base class function in derived class, then we have to declare the function in base class as **virtual**.

Q. How can we achieve multiple inheritance in C#.Net?

A. We can achieve multiple inheritance in C# through Interfaces.

Q. What are delegates and Events?

A. A delegate in C# is similar to a function pointer in C or C++. Using a delegate allows the programmer to encapsulate a reference to a method inside a delegate object. The delegate object can then be passed to code which can call the referenced method, without having to know at compile time which method will be invoked. Events are associated with Delegates. Events fires automatically when a delegate is called.

Q. What if an error occurs in Catch block, how can you prevent it?

A. We can insert a new try-catch block inside the catch block. Try-catch blocks are used for handling errors in c#.

Q. What properties we used to call stored procedure in C#?

A. `cmd.CommandType=CommandType.StoredProcedure;`
`cmd.CommandText="Name Of StoreProcedure";`
`cmd.Executenonquery();`

Q. Which collection will you use to store different types of objects collection like... int, employee, student, object etc?

A. We would use array list. Every element in an array list is of object type. Since object class is base class of all the types in .net or any custom class so it will be possible to store different types of objects like int, employee(Custom class), student(custom class), object (base class itself) into an Array list collection.

Q. What is abstraction and data hiding?

A. **Abstraction:** Abstraction refers to removal/reduction of irrelevant data or unnecessary data or confidential data from a Class.

Data Hiding: Data hiding is a feature provided by the abstraction for hiding the data from the class.

Q. What in Data Encapsulation?

A. Encapsulation is defined as the process of wrapping up the data members and member functions together into a single unit called class. It can also be defined as the concept that an object totally separates its interface from its implementation. The concept of Encapsulation hides the implementation details behind its interface.

Q. Can multiple catch blocks be executed?

A. No

Q. When an object of derived class is created which constructor will be called first (base class or derived class)?

A. Base Class Constructor

Q. How can we forcefully throw exception?

A. Using throw keyword. E.g. throw new exception (“Your Exception”)

Q. What is the use of Interface?

A. In big companies, project manager understand the client requirement and make interfaces. These interfaces are then distributed to the under developers. Since interfaces are just outline of the method, so developers must use them in their class and code them. Developers need only concentration on coding. He need not worry about which methods he uses to achieve client requirement. So interfaces provide only outline to the developer to get the goal.

Q. What is Polymorphism?

A. ‘Polymorphism’ means one name multiple forms. In it we can use the same thing for different purposes.

1) Compile Time Polymorphism.

i. Operator Overloading.

ii. Function Overloading.

2) Run Time Polymorphism.

i. Virtual Functions

Q. What is TimeSpan in C#?

A. TimeSpan is the datatype in C#.which contain day,hour,min,sec

TimeSpan myTimeSpan = new TimeSpan (2,12,0,12);

Q. Difference between string and stringbuilder?

A. System.String is immutable. System.StringBuilder was designed with the purpose of having a mutable string where a variety of operations can be performed.

Q. How to make a class non-inheritable other than sealed?

A. There are two ways:

1: Make the base class as static.

2: Declare a private constructor in the base class.

Q. Can we define a variable with the access modifier private in an interface?

A. No you cannot.

1. Because an interface cannot contain fields.

2. For methods also, you cannot give an access modifier. By default, the methods will be public

Q. What are the pillars of OOPs in c#?

A. The four pillars of oops is:
Abstraction, encapsulation, polymorphism, inheritance

Q. What are properties and indexer?

A. Using an object like an array is called Indexer. Indexer is similar to properties. Indexer is a collection of set and gets procedures. Indexer name must be "this" only. One class can have only one indexer.

1. An index is identified by its signature. But a property is identified its name.
2. An indexer is always an instance member, but a property can be static also.
3. An indexer is accessed through an element access. But a property is through a member access.

Q. What does the term immutable means?

A. It means that this object can't be changed but if you want another value to the same object another instance of the object is created and leaves the current instance unchanged. An example of immutable is STRING. String Builder is Mutable.

Q. Can interface inherits another interface?

A. Yes interface inherits interface only.

Q. I have a class declared as below

```
public class a
{
    public void add()
    {}
}
```

What is the difference between

a a1 =new a;

and simply

a a1;

A. In a a1=new a memory is allocated to the main function to execute. a a1; At this stage, a1 is of type a, but it does not actually contain the object data yet. To contain the object data, you need to use the new keyword to create a new instance of the a class, this process is known as object instantiation: a a1 = new a();

Q. What is boxing and UnBoxing?

A. Converting value type to object and vice versa.

Q. What is Serialization?

A. Serialization is process of converting an object into stream of bytes so that it can be stored or transferred across the network.

Q. What are collections in c sharp?

A. Collection Classes have the following properties:

→Collection classes are defined as part of the System.Collections or **System.Collections.Generic** namespace

→Most collection classes derive from the interfaces **ICollection**, **IComparer**, **IEnumerable**, **IList**, **IDictionary**, and **IDictionaryEnumerator** and their generic equivalents.

→Using generic collection classes provides increased type-safety and in some cases can provide better performance, especially when storing value types.

Q. How to sort array elements in descending order?

A. Elements of an array may not be sorted by default. To sort them in descending order, the Sort() method is first called. Next, to descend the order, call the Reverse() method.

Q. What are partial classes?

A. A partial class, or partial type, is a feature of some object oriented computer programming languages in which the declaration of a class may be split across multiple source-code files, or multiple places within a single file. In this Code is partially divided among different classes.