Top 52 C# Interview Questions & Answers

Here are some frequently asked interview questions for freshers as well as experienced C# developers candidates to get the right job.

1. What is C#?

C# is an object-oriented, type-safe, and managed language that is compiled by .Net framework to generate Microsoft Intermediate Language.

2. Explain types of comment in C# with examples

Single line

Example:

```
//This is a single line comment
```

ii. Multiple line (/* */)

Example:

```
/*This is a multiple line comment
We are in line 2
Last line of comment*/
```

iii. XML Comments (///).

Eg:

```
/// summary;
/// Set error message for multilingual language.
/// summary
```

3. Can multiple catch blocks be executed?

No, Multiple catch blocks can't be executed. Once the proper catch code executed, the control is transferred to the finally block, and then the code that follows the finally block gets executed.

4. What is the difference between public, static, and void?

Public declared variables or methods are accessible anywhere in the application. Static declared variables or methods are globally accessible without creating an instance of the class. Static member are by default not globally accessible it depends upon the type of access modified used. The compiler stores the address of the method as the entry point and uses this information to begin execution before any objects are created. And Void is a type modifier that states that the method or variable does not return any value.

5. What is an object?

 (/images/1/101818_0801_Top50CInter1.jpg)

An object is an instance of a class through which we access the methods of that class. "New" keyword is used to create an object. A class that creates an object in memory will contain the information about the methods, variables, and behavior of that class.

6. Define Constructors

A constructor is a member function in a class that has the same name as its class. The constructor is automatically invoked whenever an object class is created. It constructs the values of data members while initializing the class.

7. What is Jagged Arrays?

The Array which has elements of type array is called jagged Array. The elements can be of different dimensions and sizes. We can also call jagged Array as an Array of arrays.

8. What is the difference between ref & out parameters?

An argument passed as ref must be initialized before passing to the method whereas out parameter needs not to be initialized before passing to a method.

9. What is the use of 'using' statement in C#?

The 'using' block is used to obtain a resource and process it and then automatically dispose of when the execution of the block completed.

10. What is serialization?

When we want to transport an object through a network, then we have to convert the object into a stream of bytes. The process of converting an object into a stream of bytes is called Serialization. For an object to be serializable, it should implement ISerialize Interface. De-serialization is the reverse process of creating an object from a stream of bytes.

11. Can we use "this" command within a static method?

We can't use 'This' in a static method because we can only use static variables/methods in a static method.

12. What is the difference between constants and read-only?

Constant variables are declared and initialized at compile time. The value can't be changed afterward. Read-only is used only when we want to assign the value at run time.

13. What is an interface class? Give one example of it

An Interface is an abstract class which has only public abstract methods, and the methods only have the declaration and not the definition. These abstract methods must be implemented in the inherited classes.

```csharp
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace DemoApplication
{
 interface Guru99Interface
 {
  void SetTutorial(int pID, string pName);
  String GetTutorial();
 }

 class Guru99Tutorial : Guru99Interface
 {
  protected int TutorialID;
  protected string TutorialName;

  public void SetTutorial(int pID, string pName)
  {
   TutorialID = pID;
   TutorialName = pName;
  }

  public String GetTutorial()
  {
   return TutorialName;
  }

  static void Main(string[] args)
  {
   Guru99Tutorial pTutor = new Guru99Tutorial();

   pTutor.SetTutorial(1,".Net by Guru99");

   Console.WriteLine(pTutor.GetTutorial());

   Console.ReadKey();
  }
 }
}
```

14. What are value types and reference types?

A value type holds a data value within its own memory space. Example

```
int a = 30;
```

Reference type stores the address of the Object where the value is being stored. It is a pointer to another memory location.

```
string b = "Hello Guru99!!";
```

15. What are Custom Control and User Control?

Custom Controls are controls generated as compiled code (Dlls), those are easier to use and can be added to toolbox. Developers can drag and drop controls to their web forms. Attributes can, at design time. We can easily add custom controls to Multiple Applications (If Shared Dlls). So, If they are private, then we can copy to dll to bin directory of web application and then add reference and can use them.

User Controls are very much similar to ASP include files, and are easy to create. User controls can't be placed in the toolbox and dragged - dropped from it. They have their design and code-behind. The file extension for user controls is ascx.

16. What are sealed classes in C#?

We create sealed classes when we want to restrict the class to be inherited. Sealed modifier used to prevent derivation from a class. If we forcefully specify a sealed class as base class, then a compile-time error occurs.

17. What is method overloading?

Method overloading is creating multiple methods with the same name with unique signatures in the same class. When we compile, the compiler uses overload resolution to determine the specific method to be invoke.

18. What is the difference between Array and Arraylist?

In an array, we can have items of the same type only. The size of the array is fixed when compared. To an arraylist is similar to an array, but it doesn't have a fixed size.

19. Can a private virtual method can be overridden?

No, because they are not accessible outside the class.

20. Describe the accessibility modifier "protected internal".

Protected Internal variables/methods are accessible within the same assembly and also from the classes that are derived from this parent class.

21. What are the differences between System.String and System.Text.StringBuilder classes?

System.String is immutable. When we modify the value of a string variable, then a new memory is allocated to the new value and the previous memory allocation released. System.StringBuilder was designed to have a concept of a mutable string where a variety of operations can be performed without allocation separate memory location for the modified string.

22. What's the difference between the System.Array.CopyTo() and System.Array.Clone() ?

Using Clone() method, we creates a new array object containing all the elements in the original Array and using CopyTo() method. All the elements of existing array copies into another existing array. Both methods perform a shallow copy.

23. How can we sort the elements of the Array in descending order?

Using Sort() methods followed by Reverse() method.

24. Write down the C# syntax to catch an exception

To catch an exception, we use try-catch blocks. Catch block can have a parameter of system.Exception type.

Eg:

```
try {
    GetAllData();
}
catch (Exception ex) {
}
```

In the above example, we can omit the parameter from catch statement.

25. What's the difference between an interface and abstract class?

Interfaces have all the methods having only declaration but no definition. In an abstract class, we can have some concrete methods. In an interface class, all the methods are public. An abstract class may have private methods.

26. What is the difference between Finalize() and Dispose() methods?

Dispose() is called when we want for an object to release any unmanaged resources with them. On the other hand, Finalize() is used for the same purpose, but it doesn't assure the garbage collection of an object.

27. What are circular references?

Circular reference is situation in which two or more resources are interdependent on each other causes the lock condition and make the resources unusable.

28. What are generics in C#.NET?

Generics are used to make reusable code classes to decrease the code redundancy, increase type safety, and performance. Using generics, we can create collection classes. To create generic collection, System.Collections.Generic namespace should be used instead of classes such as ArrayList in the System.Collections namespace. Generics promotes the usage of parameterized types.

29. What is an object pool in .NET?

An object pool is a container having objects ready to be used. It tracks the object that is currently in use, total number of objects in the pool. This reduces the overhead of creating and re-creating objects.

30. List down the commonly used types of exceptions in .net

ArgumentException, ArgumentNullException , ArgumentOutOfRangeException, ArithmeticException, DivideByZeroException ,OverflowException , IndexOutOfRangeException ,InvalidCastException ,InvalidOperationException , IOEndOfStreamException , NullReferenceException , OutOfMemoryException , StackOverflowException etc.

31. What are Custom Exceptions?

Sometimes there are some errors that need to be handled as per user requirements. Custom exceptions are used for them and are used defined exceptions.

32. What are delegates?

Delegates are same are function pointers in C++, but the only difference is that they are type safe, unlike function pointers. Delegates are required because they can be used to write much more generic type-safe functions.

33. How do you inherit a class into other class in C#?

Colon is used as inheritance operator in C#. Just place a colon and then the class name.

public class DerivedClass : BaseClass

34. What is the base class in .net from which all the classes are derived from?

```
System.Object
```

35. What is the difference between method overriding and method overloading?

In method overriding, we change the method definition in the derived class that changes the method behavior. Method overloading is creating a method with the same name within the same class having different signatures.

36. What are the different ways a method can be overloaded?

Methods can be overloaded using different data types for a parameter, different order of parameters, and different number of parameters.

37. Why can't you specify the accessibility modifier for methods inside the interface?

In an interface, we have virtual methods that do not have method definition. All the methods are there to be overridden in the derived class. That's why they all are public.

38. How can we set the class to be inherited, but prevent the method from being over-ridden?

Declare the class as public and make the method sealed to prevent it from being overridden.

39. What happens if the inherited interfaces have conflicting method names?

Implement is up to you as the method is inside your own class. There might be a problem when the methods from different interfaces expect different data, but as far as compiler cares you're okay.

40. What is the difference between a Struct and a Class?

Structs are value-type variables, and classes are reference types. Structs stored on the Stack causes additional overhead but faster retrieval. Structs cannot be inherited.

41. How to use nullable types in .Net?

Value types can take either their normal values or a null value. Such types are called nullable types.

```
Int? someID = null;
If(someID.HasVAlue)
{
}
```

42. How we can create an array with non-default values?

We can create an array with non-default values using Enumerable.Repeat.

43. What is difference between "is" and "as" operators in c#?

"is" operator is used to check the compatibility of an object with a given type, and it returns the result as Boolean.

"as" operator is used for casting of an object to a type or a class.

44. What's a multicast delegate?

A delegate having multiple handlers assigned to it is called multicast delegate. Each handler is assigned to a method.

45. What are indexers in C# .NET?

Indexers are known as smart arrays in C#. It allows the instances of a class to be indexed in the same way as an array.

Eg:

```
public int this[int index]    // Indexer declaration
```

46. What is difference between the "throw" and "throw ex" in .NET?

"Throw" statement preserves original error stack whereas "throw ex" have the stack trace from their throw point. It is always advised to use "throw" because it provides more accurate error information.

47. What are C# attributes and its significance?

C# provides developers a way to define declarative tags on certain entities, eg. Class, method, etc. are called attributes. The attribute's information can be retrieved at runtime using Reflection.

48. How to implement a singleton design pattern in C#?

In a singleton pattern, a class can only have one instance and provides an access point to it globally.

Eg:

```
Public sealed class Singleton
{
Private static readonly Singleton _instance = new Singleton();
}
```

49. What is the difference between directcast and ctype?

DirectCast is used to convert the type of object that requires the run-time type to be the same as the specified type in DirectCast.

Ctype is used for conversion where the conversion is defined between the expression and the type.

50. Is C# code is managed or unmanaged code?

C# is managed code because Common language runtime can compile C# code to Intermediate language.

51. What is Console application?

A console application is an application that can be run in the command prompt in Windows. For any beginner on .Net, building a console application is ideally the first step, to begin with.

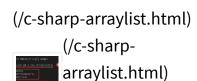## 52. Give an example of removing an element from the queue

The dequeue method is used to remove an element from the queue.

```csharp
using System;
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace DemoApplication
{
 class Program
 {
  static void Main(string[] args)
  {
   Queue qt = new Queue();
   qt.Enqueue(1);
   qt.Enqueue(2);
   qt.Enqueue(3);

   foreach (Object obj in qt)
   {
    Console.WriteLine(obj);
   }
    Console.WriteLine(); Console.WriteLine();
    Console.WriteLine("The number of elements in the Queue " + qt.Count);
    Console.WriteLine("Does the Queue contain " + qt.Contains(3));
    Console.ReadKey();
  }
 }
}
```
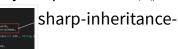
Instructions For Quiz

- This Mock Test has 15 Questions
- Each Question is of 1 marks so total 15 marks
- Multiple Choice Questions(MCQ) with only ONE valid answer

1) What is stored in Heap?*

○ int

YOU MIGHT LIKE:

C#

C#

C#

[C# ArrayList Tutorial with Examples](#)

[C# Inheritance & Polymorphism with](#)

[C# Stack with Example](#)

(/c-sharp-arraylist.html)
(/c-sharp-arraylist.html)

(/c-sharp-arraylist.html)

(/c-sharp-inheritance-polymorphism.html) (/c-sharp-inheritance-polymorphism.html)

Examples (/c-sharp-inheritance-polymorphism.html)

(/c-sharp-stack.html) (/c-sharp-stack.html)

(/c-sharp-stack.html)

C#

(/c-sharp-abstract-class.html) (/c-sharp-abstract-class.html)

C# Abstract Classes Tutorial with Example
(/c-sharp-abstract-class.html)

C#

(/c-sharp-queue.html) (/c-sharp-queue.html)

C# Queue with Examples
(/c-sharp-queue.html)

C#

(/c-sharp-access-modifiers-constructor.html) (/c-sharp-access-modifiers-constructor.html)

C# Access Modifiers(Specifiers) with Example
(/c-sharp-access-modifiers-constructor.html)

C# Tutorial

f [(https://www.facebook.com/guru99com/)](https://www.facebook.com/guru99com/)

🐦 [(https://twitter.com/guru99com)](https://twitter.com/guru99com) **in**
[(https://www.linkedin.com/company/guru99/)](https://www.linkedin.com/company/guru99/)

▶️
[(https://www.youtube.com/channel/UC19i1XD6k88KqHlET8atqFQ)](https://www.youtube.com/channel/UC19i1XD6k88KqHlET8atqFQ)

✉️
[(https://forms.aweber.com/form/46/724807646.htm)](https://forms.aweber.com/form/46/724807646.htm)

About

About Us [(/about-us.html)](/about-us.html)
Advertise with Us [(/advertise-us.html)](/advertise-us.html)
Write For Us [(/become-an-instructor.html)](/become-an-instructor.html)
Contact Us [(/contact-us.html)](/contact-us.html)

Career Suggestion

SAP Career Suggestion Tool [(/best-sap-module.html)](/best-sap-module.html)
Software Testing as a Career [(/software-testing-career-complete-guide.html)](/software-testing-career-complete-guide.html)

Interesting

eBook [(/ebook-pdf.html)](/ebook-pdf.html)
Blog [(/blog/)](/blog/)
Quiz [(/tests.html)](/tests.html)
SAP eBook [(/sap-ebook-pdf.html)](/sap-ebook-pdf.html)

Execute online

Execute Java Online [(/try-java-editor.html)](/try-java-editor.html)
Execute Javascript [(/execute-javascript-online.html)](/execute-javascript-online.html)
Execute HTML [(/execute-html-online.html)](/execute-html-online.html)
Execute Python [(/execute-python-online.html)](/execute-python-online.html)