

1. What is an abstract class in C# and how is it different from a regular class?

**Abstract class is helps to hide certain details and show only essential details to the user. We can create objects for regular classes from another class, but we can’t create object for an abstract class.**

1. How do you declare an abstract class in C#? Provide an example.

**Using *abstract* keyword,**

internal abstract class Sample

{

}

1. What is an interface in C# and how does it differ from an abstract class?

**Interface is completely an abstract class (Cannot use to create objects), which contains only abstract methods, we can’t declare fields or variables inside an interface.**

1. How do you declare an interface in C#? Provide an example.

**Using *interface* keyword,**internal interface ISample  
{

}

1. Can an abstract class implement an interface in C#? Provide an example.

**Yes,**

internal interface ISample

{

}

internal abstract class Sample1 : ISample

{

}

1. Can a class inherit from an abstract class and implement an interface at the same time? Provide an example.

internal interface ISample

{

}

internal abstract class Sample1

{

}

internal class Sample2 : Sample1, ISample

{

}

1. What happens if a class does not implement all methods of an interface it implements?

**It’ll shows a compilation error.**

1. Can an interface inherit from another interface in C#? Provide an example.

Yes,

internal interface ISample

{

}

internal interface ISample2 : ISample

{

}

1. Provide an example of a class that implements multiple interfaces in C#.

internal interface ISample1

{

}

internal interface ISample2

{

}

internal interface ISample3

{

}

internal class Sample : ISample1, ISample2, ISample3

{

}

1. How can you enforce that a method in an abstract class must be overridden in derived classes?

**Must use the *same signature* and *override* keyword**