**1) What is an Object and a Class?**

A class is an encapsulation of methods and properties, also call the foundation of C#. It is used to represent a real-time entity which brings all the instances together in a single unit.

An object is an instance of a class. Technically, object is block of memory allocated which can be stored in the form of array, variables and collections.

**2) Mention four fundamental of OOP(Object Oriented Programming) concepts.**

**Four fundamental concepts of OOP are:**

* **Encapsulation** – The Internal representation of an object is hidden from the view outside object’s definition. Only the required information can be accessed whereas the rest of the data implementation is hidden.
* **Abstraction** – It is a process of identifying the critical behavior and data of an object and eliminating the irrelevant details.
* **Inheritance** – It is the ability to create new classes from another class. It is done by accessing, modifying and extending the behavior of objects in the parent class.
* **Polymorphism** – The name means, one name, many forms. It is achieved by having multiple methods with the same name but different implementations.

**3) What is Managed and Unmanaged code?**

Managed Code:

Managed code is developed using the .Net Framework which is supported by C# or VB.Net.

It is directly executed by CLR (Common Language Runtime) and its lifecycle including object creation, memory location and object disposal managed at the Runtime. So language which are written in .Net Framework is managed code.

Unmanaged code:

Unmanaged code is any type of code which is executed by application runtime of any other framework apart from .Net Platform. Such application runtime will take care of security, memory and other performance operations.

**4) What is an Interface?**

An Interface is a class with no implementation. It contains declaration of methods, properties, and events.

**5) What are the different types of classes in C#?**

**The different types of class in C# are:**

* **Partial class** – Allows its members to be divided or shared with multiple .cs files. It is denoted by the keyword *Partial.*
* **Sealed class** – It is a class which cannot be inherited. To access the members of a sealed class, we need to create the object of the class.  It is denoted by the keyword *Sealed*.
* **Abstract class** – It is a class whose object cannot be instantiated. The class can only be inherited. It should contain at least one method.  It is denoted by the keyword *abstract.*
* **Static class** – It is a class which does not allow inheritance. The members of the class are also static.  It is denoted by the keyword *static*. This keyword tells the compiler to check for any accidental instances of the static class.

**6) Explain Code compilation in C#.**

**There are four steps in code compilation which include:**

* Compiling the source code into Managed code by C# compiler.
* Combining the newly created code into assemblies.
* Loading the Common Language Runtime(CLR).
* Executing the assembly by CLR.