**Q1. All about properties in C#.**

* When we want to access the private members/class from outside the class then we can use property to access it.
* Properties can be read-only or write-only.

Property has two methods:-

**Get** :- Get is use to get the value of the variable.

If get is a function then return type is the property type

**Set :-** Set is use to set the value of variable.

If set is a function then return type is Void

**Code Snippet:-**

class Person

{

private string name;

public string Name // property

{

get { return name; } // get method

set { name = value; } // set method

}

}

**Explanation:-**

Here,

* Name property is associated with the name field.
* The get method returns the value of the variable name.
* The set method assigns a value to name variable.

**Auto complete Property:-**

Public int Age

{

get;

set;

}

Here, both get and set are auto complete Property.

**Q2. All about Constructors in C#.**

* Constructor is a special method that is used to initialize objects
* Constructor is use to Create Object and initialize it.
* Constructor invoke automatically at the time of object creation.
* There is no return type of a constructors

**Types of Constructor are:-**

1. **Default Constructor:-**

* There is a default constructor if no constructor is defined by user
* It automatically invoked at the time of object creation

1. **Parameter less Constructor:-**

* Constructor which has no argument is known as Parameter less constructor.
* It is invoked at the time of creating object.

1. **Parameterized Constructor:-**

* Constructor which has parameters is called parameterized constructor.
* It is used to provide different values to different objects.

1. **Static Constructor:-**

* It is the constructor that should be called before the first object created.
* It is only called once
* We cannot overload static constructors
* Only single static constructor can be used in a class
* We can’t access none static values in constructor.

1. **Private Constructor:-**

* These are used to prevent creating Instances of a class when there are no instance fields or methods, such as Math class

1. **Copy Constructor:-**

* Constructor that takes reference of an another object
* It copies the data of one object into another object

Public person (person p)

{

Name = p.name;

Age = p.age;

}

Person p2 = new person (p);

Here person p object copies the data into person p2 object

**Q2. Virtual, Override, New and Sealed keywords.**

**1. Virtual:-**

* In this the parent’s child have to rewrite its methods.
* It uses methods

**2. Override:-**

* In this the child class can overwrite parent’s methods.
* It also uses methods

**3. New:-**

**4. Sealed:-**

* Sealed keyword apply restrictions on the class and method
* Child can’t use hierarchical parent’s method.
* Only can use its parent’s methods
* It uses class