**Name: Ram Gopal**

**1. Explain about all properties in brief.**

In C# programming language, property is a data field, which has get and set methods.

Properties are called members of classes and interfaces.

There are various types of properties based on the get and set, which are given below...

* **Read-Write Properties:**

When property contains both get and set methods.

Example: 1

Public class Person

{

String name {get; set;}

Int age {get; set}

}

* **Read-Only Properties:**

When property contains only get method, this type of property called read-only property.

Example: 2

Public class Person

{

String name{get;}

}

* **Write-Only Properties:**

When property contains only set method, this type of property called write-only property.

Example: 3

Public class Person

{

string Name{set;}

}

* **Auto Implemented Properties:**

Auto implemented properties contain both get and set methods without additional logic.

Example: 4

Public class Person

{

Private string Name;

Public string Name {

get=>name;

set=>name=value;

}

}

**2. Explain about all constructor's type.**

In object-oriented programming, is a special type of method which is used to construct and initialize an object. Constructor is automatically called whenever we create an object. The name of the constructor must be same as of its class name. Constructor does not have any return type explicitly.

In C# programming language, there are various types of constructor. Which are given as below.

* **Default Constructor**
* **Parameterized Constructor**
* **Static Constructor**
* **Private Constructor**
* **Copy Constructor**

**Default Constructor:**

Default constructor is a type of constructor, which does not have any parameters. If we do not create default constructor explicitly, the class will automatically create default constructor when an object is created.

Example: 5

**…. Before Object Creation**

public class Test

{

}

…. **After Object Creation**

public class Test

{

public Test ()

{

}

}

**Parameterized Constructor:**

Parameterized constructor is a type of constructor, which accepts parameters as input. When we initialize properties of objects at the time of object creation, then we use parameterized constructor.

Example: 6

public class Test

{

public Test (int age, string name)

{

}

}

The above constructor takes two arguments at the object creation.

**Static Constructor:**

Static constructor is a type of constructor, which is invoked before first object is created. Static constructor must be parameter less. It means, one class can have only one static constructor. We cannot access non-static field, non-static methods and non-static properties inside static constructor.

**Private Constructor:**

In case of private constructor, we cannot create object of that class. We use private constructor for security purposes.

**Copy Constructor:**

The constructor, which takes parameter as object of its own class is called copy constructor. The copy constructor is use to create object on the basis of existing objects.

Example: 7

public Person (Person p)

{

Name =p.name;

Age =p.age;

}

**3. Define virtual keyword, override keyword, and new keyword.**

**Virtual Keyword:**

Virtual is a keyword in C# programming language, which is used with methods in parent class. When class inherits virtual methods, child class redefines with override keyword.

Example: 8

Public class Test

{

Public void virtual display () {

//

}

}

**Override Keyword:**

Override is a keyword which is used with methods in child class. It is used to redefine virtual methods of parent class.

Example: 9

Public class West

{

Public void override display () {

//

}

}

**Sealed Keyword:**

The sealed keyword is used with method to restrict the method to be overridden in child class.

**New Keyword:**

New is a keyword special keyword in C#, which is used hide parent class method