Feb 15th Assignment

By

Chandolu Surya Teja

1. Write what is assembly in C#

Assembly:

An Assembly is a basic building block of .Net Framework applications. It is basically a compiled code that can be executed by the CLR. An assembly is a collection of types and resources that are built to work together and form a logical unit of functionality.

An Assembly can be a DLL or exe depending upon the project that we choose.

Assemblies are only loaded into memory if they are required. If they aren't used, they aren't loaded. This means that assemblies can be an efficient way to manage resources in larger projects.

2. In a tabular format write the access modifiers and explain

Table:

	Same Assembly			Different Assembly	
	Base Class	Derived Class	Outer Class	Derived Class	Outer Class
Public	Yes	Yes	Yes	Yes	Yes
Private	Yes	No	No	No	No
Protected	Yes	Yes	No	Yes	No
Internal	Yes	Yes	Yes	No	No
Protected Internal	Yes	Yes	Yes	Yes	No

Code:

Same Assembly:

```csharp
using System;

namespace AccessModifiers
{
    /// <summary>
    /// Base Class
    /// </summary>
    public class BaseClass
    {
        public int a;
        private int b;
        protected int c;
        internal int d;
        protected internal int e;
        /// <summary>
        /// Base Class Method
        /// </summary>
        public void BaseClassMethod()
        {
```

```csharp
                a = 10;
                b = 20;
                c = 30;
                d = 40;
                e = 50;
            }
        }
        /// <summary>
        /// Derived Class
        /// </summary>
        public class DerivedClass : BaseClass
        {
            /// <summary>
            /// Derived Class Method
            /// </summary>
            public void DerivedClassMethod()
            {
                a = 10;
                //b = 20; Private variable cannot accessble in same derived class
                c = 30;
                d = 40;
                e = 50;
            }
        }
        /// <summary>
        /// Main Class
        /// </summary>
        public class MainClass
        {
            /// <summary>
            /// Base Class Object Created
            /// </summary>
            BaseClass bc = new BaseClass();
            /// <summary>
            /// Main Class Method
            /// </summary>
            public void MainClassMethod()
            {
                bc.a = 10;
                //bc.b = 20; Private variable cannot accessble in same main class
                //bc.c = 30; Protected variable cannot accessble in same main class
                bc.d = 40;
                bc.e = 50;
            }
        }
    }


Different Assembly:


using System;
using AccessModifiers;

namespace AccessModifiers2
{
```

```csharp
/// <summary>
/// Other Derived Class
/// </summary>
public class OtherDerivedClass : BaseClass
{
    /// <summary>
    /// Other Derived Class Method
    /// </summary>
    public void OtherDerivedClassMethod()
    {
        a = 10;
        //b = 20; Private variable cannot accessble in other derived class
        c = 30;
        //d = 40; Internal variable cannot accessble in other derived class
        e = 50;
    }
}
/// <summary>
/// Other Main Class
/// </summary>
public class OtherMainClass
{
    BaseClass bc2 = new BaseClass();
    /// <summary>
    /// Other Main Class Method
    /// </summary>
    public void OtherMainClassMethod()
    {
        bc2.a = 10;
        //bc2.b = 20; Private variable cannot accessble in other main class
        //bc2.c = 30; Protected variable cannot accessble in other main class
        //bc2.d = 40; Internal variable cannot accessble in other main class
        //bc2.e = 50; Protected Internal variable cannot accessble in other
main class
    }
}
}
```