

Q1.Research and write what is Assembly in C#?

An assembly is a file that is automatically generated by the compiler upon successful compilation of every .NET application like ddl or exe.

Assemblies are two types

1.Private Assembly

2.Shared Assembly

1.Private Assembly:

It is an assembly that is used by a single application only.

If we have a project, in which we refer to a dll so when we build that project that dll will be copied in the bin folder of our project. That dll becomes a private assembly within our project.

2.Shared Assembly:

These are used in more than one project.

These are installed in GAC.

Assemblies that are installed in GAC are made available to all the .Net applications on that machine.

GAC: Global Assembly Cache

It is a memory that is used to store the assemblies that are meant to be used by various applications.

Q2. In a tabular format write the  
access modifiers and explain

(as I did in the class, create two assemblies with 3 classes in first assembly, 2 classes in other assembly)

1.Sudha Assembly:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace SudhaLibrary
{
    /*****
     * Author:Sudha Kumari Sugasani
     * Purpose:Example to understand about Access Modifiers
     * *****/

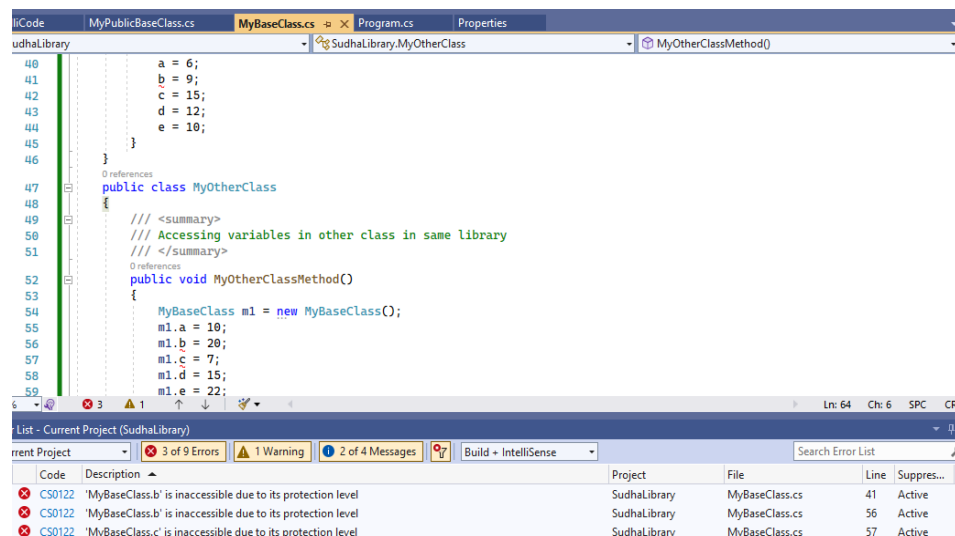
    public class MyBaseClass
    {
        public int a ;
        private int b ;
        protected int c ;
        internal int d;
        protected internal int e ;
        /// <summary>
        /// Accessing variables in same class in same library
        /// </summary>
        public void MyBaseClassMetod()
        {
            a = 5;
```

```

        b = 19;
        c = 7;
        d = 5;
        e = 15;
    }
}
public class MyDerivedClass:MyBaseClass
{
    /// <summary>
    /// Accessing variabes in derived class in same library
    /// </summary>
    public void MyDerivedClassmethod()
    {
        a = 6;
        b = 9;
        c = 15;
        d = 12;
        e = 10;
    }
}
public class MyOtherClass
{
    /// <summary>
    /// Accessing variables in other class in same library
    /// </summary>
    public void MyOtherClassMethod()
    {
        MyBaseClass m1 = new MyBaseClass();
        m1.a = 10;
        m1.b = 20;
        m1.c = 7;
        m1.d = 15;
        m1.e = 22;
    }
}
}

```

#### Errors Screenshot:



### PublicClassLibrary:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using SudhaLibrary;

namespace PublicLibrary
{
    public class MyDerivedClass:MyBaseClass
    {
        /// <summary>
        /// Implementing Base class variables into Derivedclass in other
library
        /// </summary>
        public void MyPublicBaseClassMethod()
        {
            a = 5;
            b = 10;
            c = 15;
            d= 20;
            e = 25;
        }
    }
    public class MyPublicOtherClass
    {
        /// <summary>
        /// Accessing base class variables in other class in other library
        /// </summary>
        public void MyOtherClassMethod()
        {
            MyBaseClass mb = new MyBaseClass();
            mb.a = 6;
            mb.b = 5;
            mb.c = 78;
            mb.d = 55;
            mb.e = 15;
        }
    }
}
```

### Error Screenshot:

The screenshot shows the Visual Studio IDE with the 'PublicLibrary' project selected. The code editor displays the 'MyPublicOtherClass' class, specifically the 'MyOtherClassMethod()' method. The code is as follows:

```
public void MyPublicBaseClassMethod()
{
    a = 5;
    b = 10;
    c = 15;
    d= 20;
    e = 25;
}

public class MyPublicOtherClass
{
    /// <summary>
    /// Accessing base class variables in other class in other library
    /// </summary>
    public void MyOtherClassMethod()
    {
        MyBaseClass mb = new MyBaseClass();
        mb.a = 6;
        mb.b = 5;
        mb.c = 78;
        mb.d = 55;
        mb.e = 15;
    }
}
```

The Error List at the bottom shows 6 of 9 errors, all of which are CS0122: 'MyBaseClass.b' is inaccessible due to its protection level. The errors are listed as follows:

Code	Description	Project	File	Line	Suppres...
CS0122	'MyBaseClass.b' is inaccessible due to its protection level	PublicLibrary	MyPublicBaseClass.cs	18	Active
CS0122	'MyBaseClass.b' is inaccessible due to its protection level	PublicLibrary	MyPublicBaseClass.cs	33	Active
CS0122	'MyBaseClass.c' is inaccessible due to its protection level	PublicLibrary	MyPublicBaseClass.cs	34	Active
CS0122	'MyBaseClass.d' is inaccessible due to its protection level	PublicLibrary	MyPublicBaseClass.cs	20	Active
CS0122	'MyBaseClass.d' is inaccessible due to its protection level	PublicLibrary	MyPublicBaseClass.cs	35	Active
CS0122	'MyBaseClass.e' is inaccessible due to its protection level	PublicLibrary	MyPublicBaseClass.cs	36	Active

Table to understand about Access Modifiers :

	Within	the	Assembly	Other	Assembly
	Within Class	Derived Class	Other Class	Derived Class	Other Class
Public	Yes	Yes	Yes	Yes	Yes
Private/Default	Yes	No	No	No	No
Protected	Yes	Yes	No	Yes	No
Internal	Yes	Yes	Yes	No	No
Protected Internal	Yes	Yes	Yes	Yes	No

Points to understand about Access Modifiers:

- Public can be accessed anywhere.
- Private/Default can be accessed only within the class in same assembly(library).
- Protected can be accessed within the class, derived class in both assemblies(libraries).
- Internal can be accessed within the Assembly(library).
- Protected Internal can be accessed within the assembly and can be accessed in the derived class in other assembly(library).