|  |
| --- |
| **Day-17 assignment**  **By**  **Bhanu Rama Krishna Prakash Jakkamsetti**  **15/2/2022** |

|  |
| --- |
| 1.Research and write what is assembly in c#. |
| 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 what you choose.  Assemblies are basically the following two types:  1.Private assembly  2.Shared assembly  1.Private assembly:  It is an assembly that is being used by a single application only. Suppose we have a project in which we refer to a DLL so when we build that project that DLL will be copied to the bin folder of our project. That DLL becomes a private assembly within our project. Generally, the DLLs that are meant for a specific project are private assemblies.  2.Shared assembly:  Assemblies that can be used in more than one project are known to be shared assembly. Shared assemblies are generally installed in the GAC. Assemblies that are installed in the GAC are made available to all the .NET applications on that machine.  However, there are two more types of assemblies in .NET, satellite assembly, and share assembly. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 2. In a tabular format write the access modifiers and explain. | | | | | |
|  | Within Assembly | | | Other Assembly | |
|  | Same class | Derived class | Other class | Derived class | Other 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 |
| Internal protected | YES | YES | YES | YES | NO |

|  |
| --- |
| Code: |
| Bhanulibrary:  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Bhanulibrary  {  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  \* author:bhanu ram krishna prakash jakkamsetti  \* purpose:know use of access specifiers  \* \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  public class Sameclass  {  public int a;  private int b;  protected int c;  internal int d;  internal protected int e;  public void Sameclassmethod()  {  a = 5;  b = 6;  c = 7;  d = 8;  e = 9;  }  }  class DerivedclassinSameAssembly : Sameclass  {  public void DerivedclassinSameAssemblymethod()  {    a = 5;  b = 6;  c = 7;  d = 8;  e = 9;    }  }  class OtherclassinSameAssembly  {  public void OtherclassinSameAssemblymethod()  {  Sameclass s=new Sameclass();    s. a = 5;  s. b = 6;  s. c = 7;  s. d = 8;  s. e = 9;    }  }  }  Public library:  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using Bhanulibrary;  namespace Publiclibrary  {  public class DerivedclassinOtherAssembly : Sameclass  {  public void DerivedclassinOtherAssemblymethod()  {  a = 5;  b = 6;  c = 7;  d = 8;  e = 9;  }  }  class OtherclassinOtherAssembly  {  public void OtherclassinOtherAssemblymethod()  {  Sameclass s = new Sameclass();  s.a = 5;  s.b = 6;  s.c = 7;  s.d = 8;  s.e = 9;  }  }  } |
| Output: |
| Same class same assembly    Derived class same assembly    Other class same assembly    Derived class other assembly    Other class other assembly |