

```
1 using System;
2
3 // NAME: ALIBROWN JULIUS EMMANUEL
4 // MATRIC NUMBER: 180805033
5 // COURSE: CSC 322
6 // ASSIGNMENT: ACCESS MOODIFIER
7
8 namespace MyNamespace
9 {
10     internal class Program
11     {
12         public static void Main(string[] args)
13         {
14             DerivedProgram.MyStaticMethod();
15             DerivedProgram derivedProgram = new DerivedProgram();
16             derivedProgram.MyAbstractMethod();
17             derivedProgram.MyInternalMethod();
18             derivedProgram.MyVirtualMethod();
19         }
20     }
21
22     // Declare a partial class named MyClass
23     internal abstract partial class MyClass
24     {
25         // Declare a private static field
26         private static int myStaticField;
27
28         // Declare a protected virtual method
29         public virtual void MyVirtualMethod()
30         {
31             Console.WriteLine("MyClass.MyVirtualMethod() called.");
32         }
33
34         // Declare an abstract method
35         public abstract void MyAbstractMethod();
36
37         // Declare a public static method
38         public static void MyStaticMethod()
39         {
40             Console.WriteLine("MyClass.MyStaticMethod() called.");
41         }
42     }
43
44     // Declare the rest of the partial class
45     partial class MyClass
46     {
47         // Declare an internal method that calls the protected virtual method
48         internal void MyInternalMethod()
```

```
49     {
50         MyVirtualMethod();
51     }
52
53 }
54
55 class DerivedProgram : MyClass
56 {
57     // Implement the abstract method
58     public override void MyAbstractMethod()
59     {
60         Console.WriteLine("DerivedProgram.MyAbstractMethod() called.");
61     }
62
63     public override void MyVirtualMethod()
64     {
65         Console.WriteLine("MyClass.MyVirtualMethod() called.");
66     }
67 }
68 }
69
```