Day 10 Morning Assignments

By

Praveen Chakravarthi

04-02-2022

NB Health Care

1. Write the two points discussed about inheritance in the class

Inheritance is the process of reusing base class methods in the derived class.

Inheritance main goal is : reusability and to remove duplicate code

Inheritance types :
Single inheritance
Multiple inheritance
Multilevel inheritance

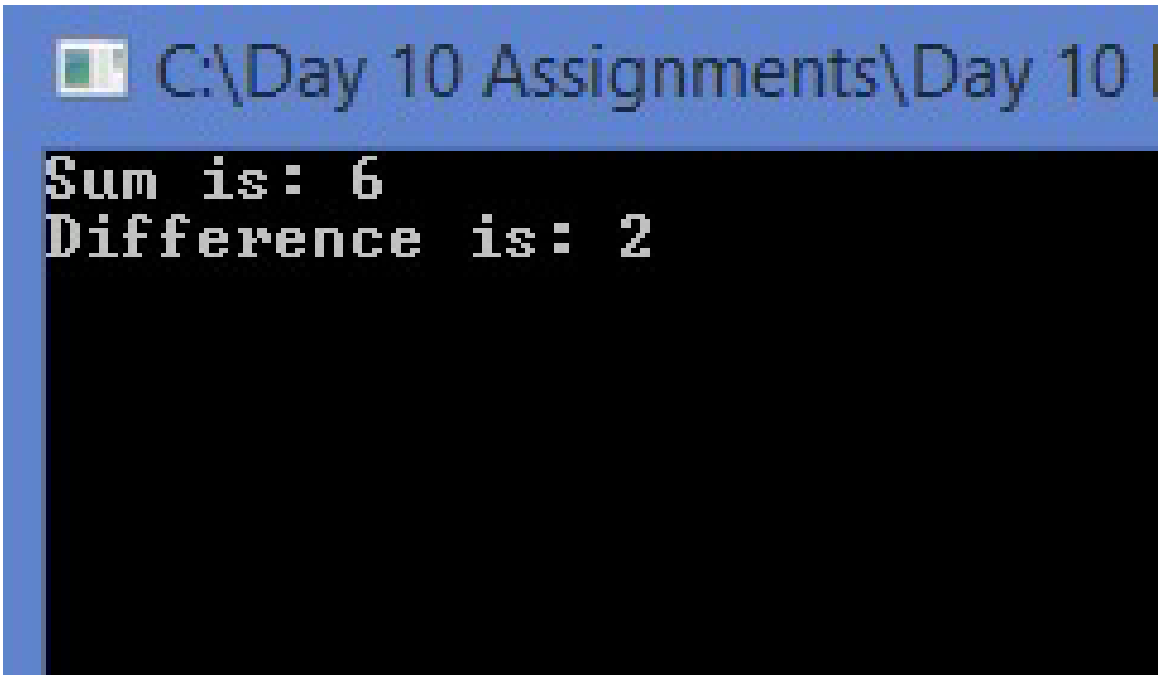
2. Write example code for
a. Single inheritance

Code:

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

namespace Day_10_Project_1
{
    // Author : Praveen Chakravarthi
    // Purpose : Example Code for Single Inheritance

    class Maths1
    {
        /// <summary>
        /// This Method Adds the given Numbers
        /// </summary>
        /// <returns></returns>
        public int Add(int a, int b)
        {
            return a + b;

        }
    }

    // Inheritance
    class Maths2 : Maths1  // Maths 2 is Child class whereas Maths1 is Parent Class
    {
        /// <summary>
        /// This Method Subtracts the given Numbers
        /// </summary>
```

```
        /// <returns></returns>
        public int Sub(int a, int b)
        {
            return a - b;
        }
    }

    internal class Program
    {
        static void Main(string[] args)
        {
            Maths2 data = new Maths2();
            Console.WriteLine($"Sum is: {data.Add(2,4)}");
            Console.WriteLine($"Difference is: {data.Sub(4,2)}");
            Console.ReadLine();

        }
    }
}
```

Output:



b. Multi level inheritance

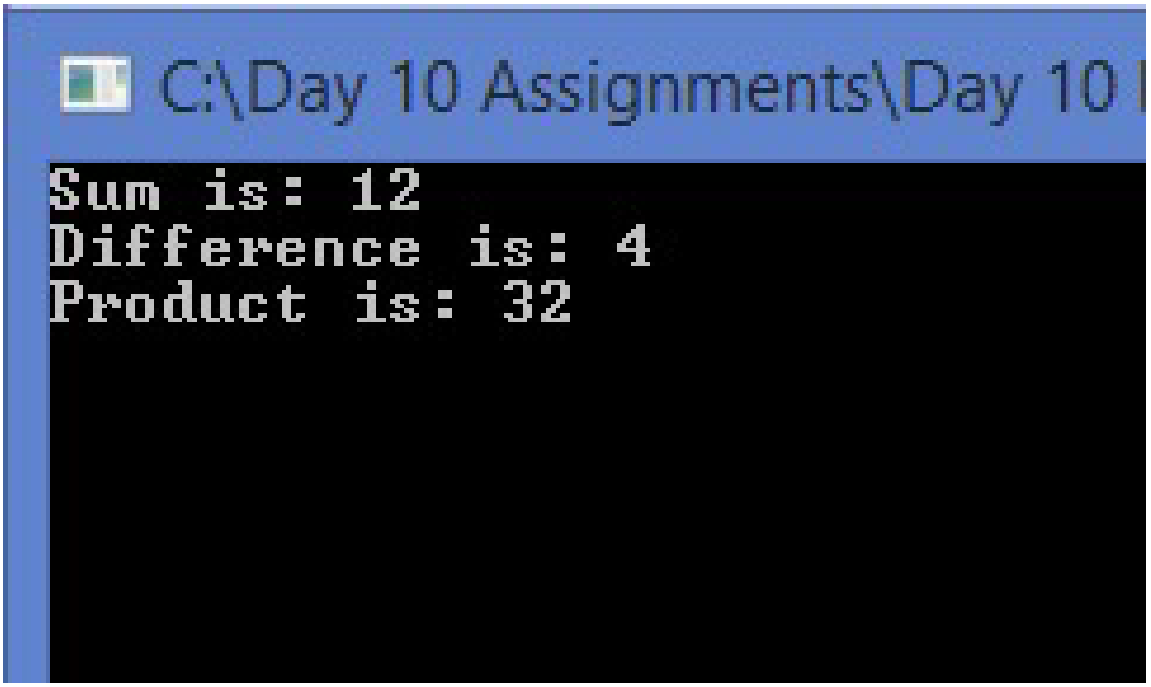
Code:

```
using System;
```

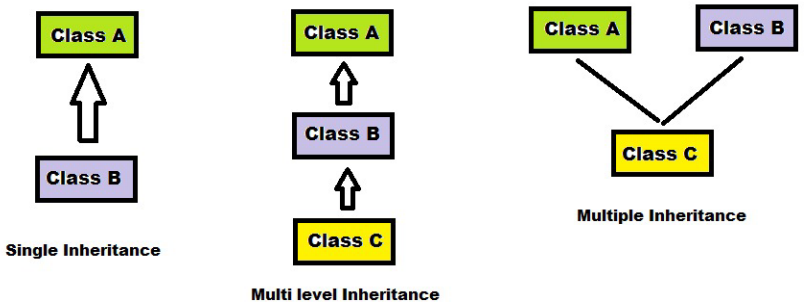
```
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace Day_10_Project_2
{
    // Author : Praveen Chakravarthi
    // Purpose : Example Code for Multilevel Inheritance
    class Maths1
    {
        /// <summary>
        /// This Method Adds the given Numbers
        /// </summary>
        /// <returns></returns>
        public int Add(int a, int b)
        {
            return a + b;

        }
    }

    // Inheritance
    class Maths2 : Maths1  // Maths 2 is Child class whereas Maths1 is Parent Class
    {
        /// <summary>
        /// This Method Subtracts the given Numbers
        /// </summary>
        /// <returns></returns>
        public int Sub(int a, int b)
        {
            return a - b;
        }
    }

    // Multilevel Inheritance
    class Maths3 : Maths2   // Maths 3 is Child class whereas Maths2 is Parent Class
    {
        /// <summary>
        /// This Method Multiplies the given Numbers
        /// </summary>
        /// <returns></returns>
        public int Mul(int a, int b)
        {
            return a* b;
        }
    }
    internal class Program
    {
        static void Main(string[] args)
        {
            Maths3 data = new Maths3();
            Console.WriteLine($"Sum is: {data.Add(8,4)}");
            Console.WriteLine($"Difference is: {data.Sub(8,4)}");
```

```
            Console.WriteLine($"Product is: {data.Mul(8,4)}");

            Console.ReadLine();
        }
    }
}
```

Output:



3. Pictorially represent 3 types of inheritance discussed in the class



4. Why multiple inheritance is not supported for classes in C#

Multiple Inheritance is not supported in C# because adding multiple inheritance added too much complexity and less benefits, induce unpredictable bugs and difficult to debug

5. What is polymorphism?

Polymorphism:

The ability of an object to take many forms
It is of 2 types :
1. Method Overloading
2. Method OverRiding

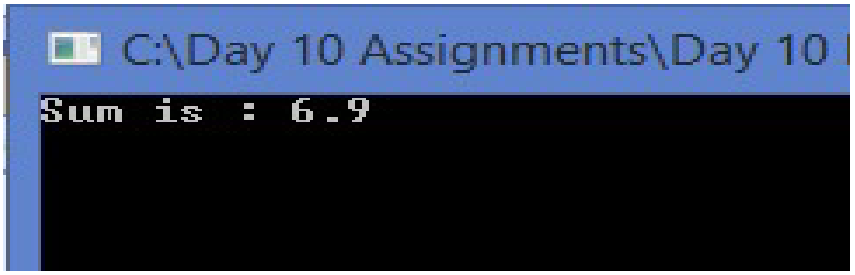
6. Write sample code for method overloading

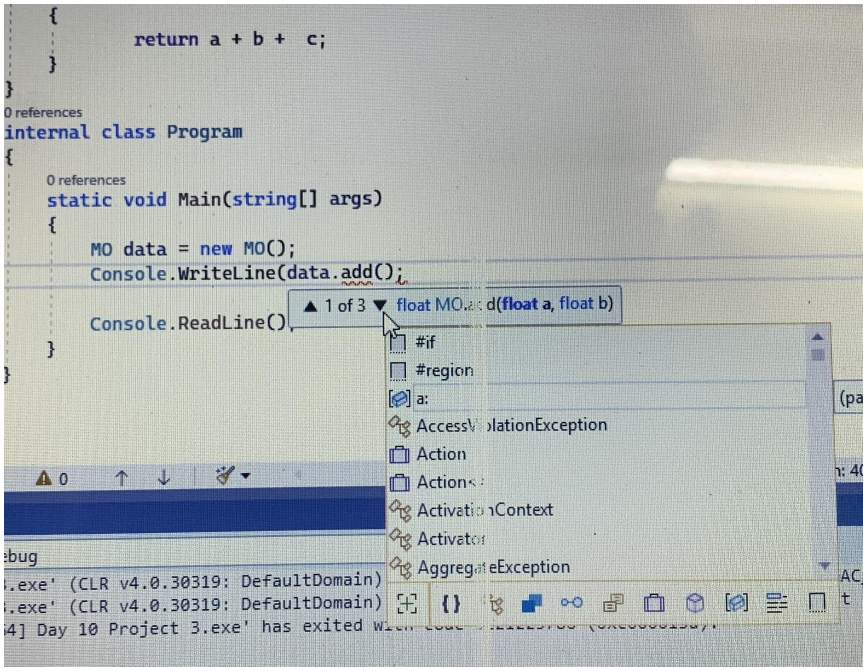
Code:

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

namespace Day_10_Project_3
{
    // Author : Praveen Chakravarthi
    // Purpose : Sample Code for Method Overloading

    class MO
    {
        public int Add(int a, int b)
        {
            return a + b;
        }

        public float Add(float a, float b)
        {
```

```
            return a + b;
        }

        public int Add(int a, int b, int c)
        {
            return a + b +  c;
        }
    }
    internal class Program
    {
        static void Main(string[] args)
        {
            MO data = new MO();
            Console.WriteLine($"Sum is : {data.Add(2.4f,4.5f)}");

            Console.ReadLine();
        }
    }
}
```

Output:





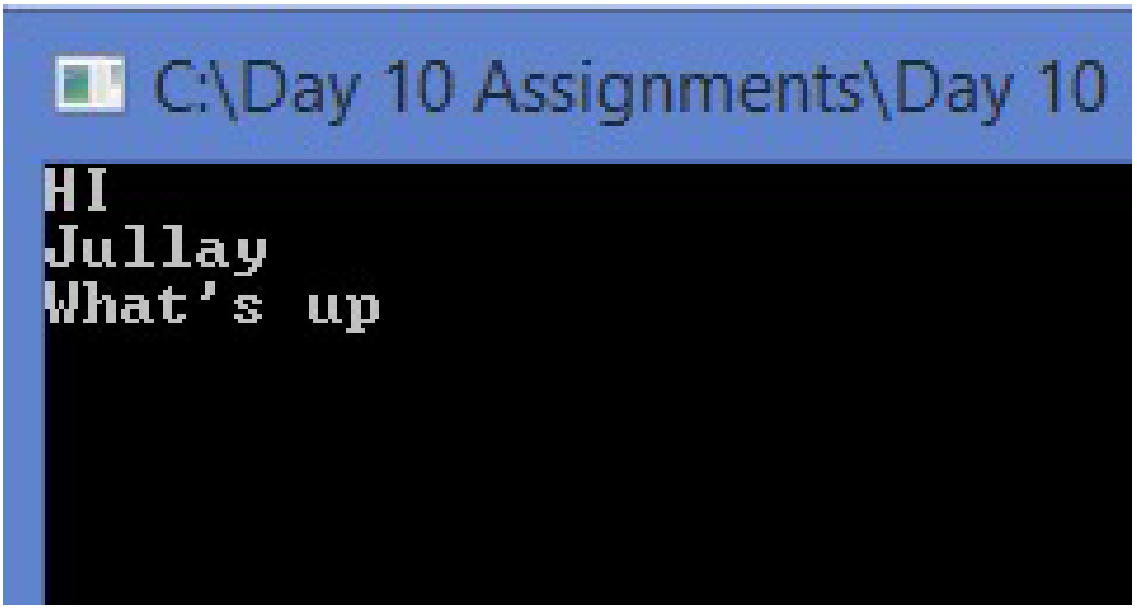
7. Write sample code for method overriding using new key word

Code:

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

namespace Day_10_Project_4
{
    // Author : Praveen Chakravarthi
    // Purpose : Sample Code for Method OverRiding

    class Hyderabad
    {
        public void PrintHi()
        {
            Console.WriteLine("HI");
        }

        public void PrintHello()
        {
            Console.WriteLine("Hello");
        }

        public void PrintWU()
        {
            Console.WriteLine("What's up");
        }
    }

    class Ladakh : Hyderabad
    {
        public new void PrintHello()
        {
            Console.WriteLine("Jullay");
        }
    }


    internal class Program
    {
        static void Main(string[] args)
        {
            Ladakh data = new Ladakh();
            data.PrintHi();
            data.PrintHello();
            data.PrintWU();
```

```
            Console.ReadLine();
        }
    }
}
```

Output:



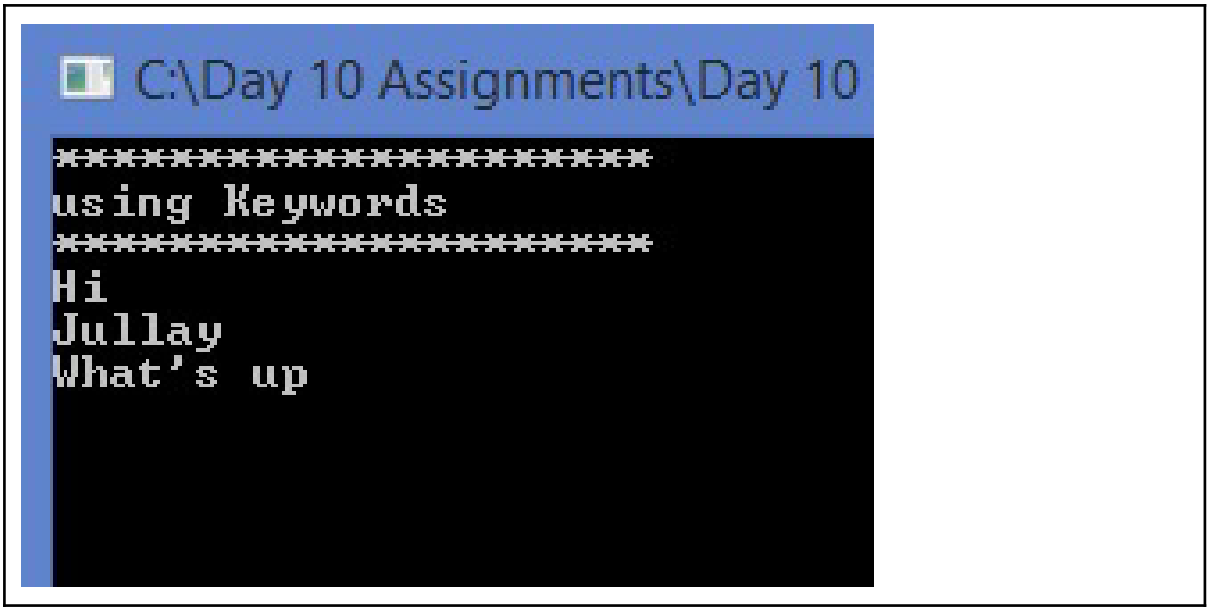
8. Research and write sample code for method overriding using virtual, override keyword.

Code:

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

namespace Day_10_Project_5
{
    // Author : Praveen Chakravarthi
    // Purpose : Sample Code for Method OverRiding using keywords

    class Hyderabad
    {
        public void PrintHi()
        {
            Console.WriteLine("Hi");
        }
```

```
        public virtual void PrintHello()
        {
            Console.WriteLine("Hello");
        }

        public void PrintWU()
        {
            Console.WriteLine("What's up");
        }
    }

    class Ladakh : Hyderabad
    {
        public override void PrintHello()
        {
            Console.WriteLine("Jullay");
        }
    }
    internal class Program
    {
        static void Main(string[] args)
        {
            Ladakh data = new Ladakh();

            Console.WriteLine("**********************");
            Console.WriteLine("using Keywords");
            Console.WriteLine("**********************");
            data.PrintHi();
            data.PrintHello();
            data.PrintWU();


            Console.ReadLine();
        }
    }
}
```

Output:


C:\Day 10 Assignments\Day 10
****
using Keywords
****
Hi
Jullay
What's up