Day 10 Morning Assignment
By
Vinay Kudali
04-02-22

NB Healthcare Technologies

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

1. Inheritance is the Process of acquiring the properties from super class to sub class.
2. The main target of the inheritance is Re-usability, and to avoid duplicate code.
3. In Inheritance, we have 4 types:
   1) Single Level Inheritance
   2) Multi level Inheritance
   3) Hierarchical Inheritance
   4) Multiple Inheritance

2. Write example code for: b. Multi level inheritance

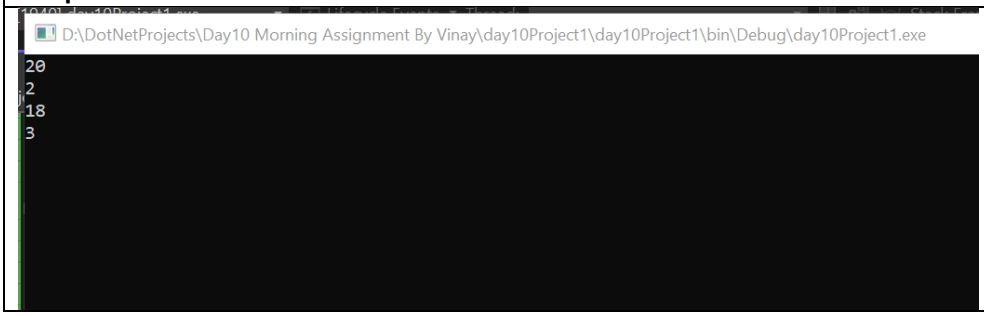
Code:

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

namespace Day10Project1
{
   //Author: Vinay Kudali
   //Purpose: Multi Level inheritance
   class AddSub

   {
      /// <summary>
      ///This method will perform addition
      /// </summary>
      /// <param name="p"></param>
      /// <param name="q"></param>
      /// <returns>Add</returns>
```

```
    public int Add(int p, int q)
    {
        return p + q;
    }
    /// <summary>
    /// This method will perform Substraction
    /// </summary>
    /// <param name="p"></param>
    /// <param name="q"></param>
    /// <returns>Substract</returns>
    public int Substract(int p, int q)
    {
        return p - q;
    }

}
class Multiply : AddSub
{
    /// <summary>
    /// This method will perform Multiplication
    /// </summary>
    /// <param name="p"></param>
    /// <param name="q"></param>
    /// <returns>mul</returns>
    public int Mul(int p, int q)
    {
        return p * q;
    }
}
class Division : Multiply
{
    /// <summary>
    /// this method will perform Division
    /// </summary>
    /// <param name="p"></param>
    /// <param name="q"></param>
    /// <returns>Div</returns>
    public int Div(int p, int q)
    {
        return p / q;
    }
}
class Program
{
    static void Main(string[] args)
    {
        Division n1 = new Division();
        Console.WriteLine(n1.Add(12, 8));
```

```
            Console.WriteLine(n1.Substract(9, 7));
            Console.WriteLine(n1.Mul(6, 3));
            Console.WriteLine(n1.Div(9, 3));
            Console.ReadLine();
        }
    }
}
```

Output:



2.a) Single Level Inheritance:

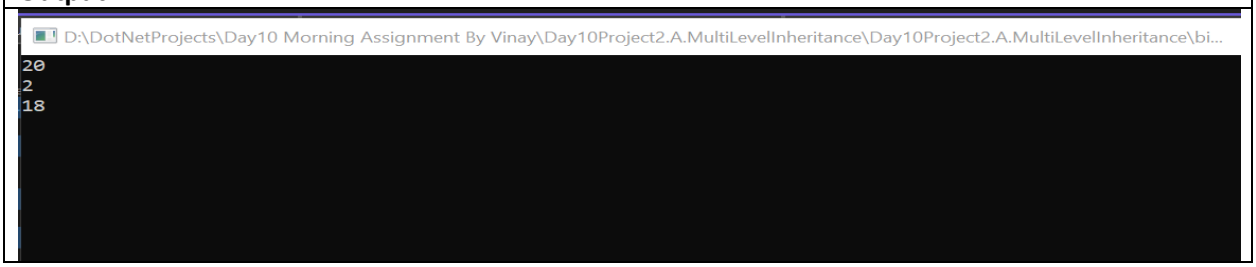
Code:

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

namespace Day10Project1

{

    //Author: Vinay Kudali
    //Purpose: Single Level inheritance

    class AddSub
    {

        /// <summary>
        ///This method will perform addition
        /// </summary>
        /// <param name="p"></param>
        /// <param name="q"></param>
        /// <returns>Add</returns>

        public int Add(int p, int q)

        {
```

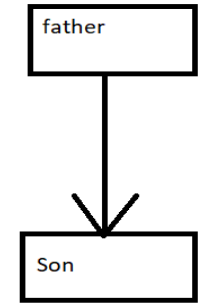
```
        return p + q;
    }

    /// <summary>
    /// This method will perform Substraction
    /// </summary>
    /// <param name="p"></param>
    /// <param name="q"></param>
    /// <returns>Substract</returns>

    public int Substract(int p, int q)

    {
        return p - q;
    }

}

class Multiply : AddSub

{
    /// <summary>
    /// This method will perform Multiplication
    /// </summary>
    /// <param name="p"></param>
    /// <param name="q"></param>
    /// <returns>mul</returns>

    public int Mul(int p, int q)

    {
        return p * q;
    }

}

class Program

{

    static void Main(string[] args)

    {

        Multiply n1 = new Multiply();
        Console.WriteLine(n1.Add(12, 8));
        Console.WriteLine(n1.Substract(9, 7));
        Console.WriteLine(n1.Mul(6, 3));
```

```
        Console.ReadLine();

      }

   }

}
```

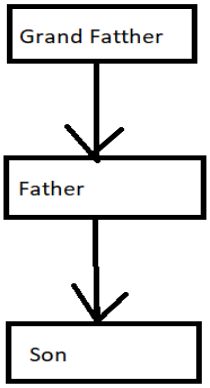
Output:



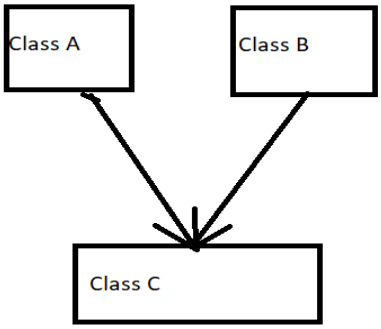
3. Pictorially represent 3 types of inheritance discussed in the class.



Single Level Inheritance



Multi Level Inheritance



Multiple inheritance

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

In Multiple inheritance, we have More than one super class So that, Sub class will get confused to acquire the properties from both super classes. Because there are two Super classes. That's why it's multiple inheritance is not supported in C#.

5. What is Polymorphism.

Polymorphism is a Single Entity Which is having multiple forms.