Day 10
Assignment
By
M Mary Margarette

Write the two points discussed about inheritance in the class.

→ Inheritance is the process of re-using parent class methods in derived class.

→ Inheritance is used to remove duplicate code.

→ Main goal of Inheritance is Re-usability of class.

---

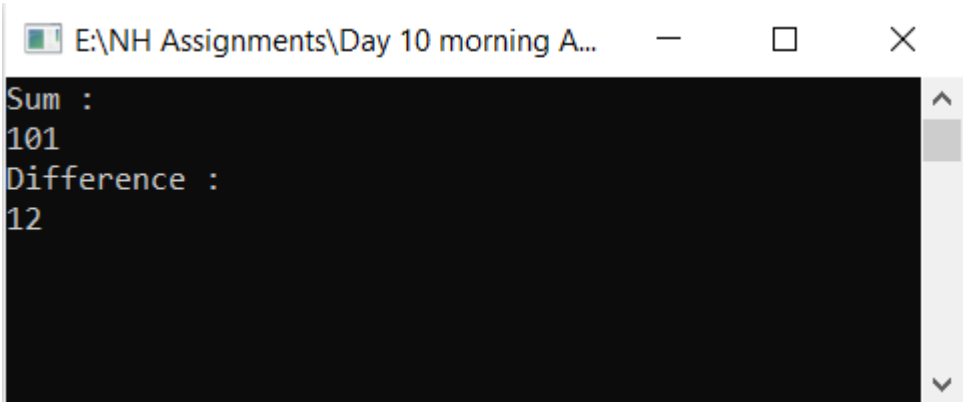
2. Write example code for: a. Single inheritance b. Multi level inheritance

a. Code for Single Inheritance

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace Day10Project1
{
   //Author :Mary Margaret
   //Purpose : Single inheritance
   class ADD
   {
      /// <summary>
      /// This method finds sum of 3 numbers
      /// </summary>

      public int Sum(int a, int b, int c)
      {
         return a + b + c;
      }

   }
   //child class
```

```
    class SUB : ADD
    {
        /// <summary>
        /// This method find Difference
        /// </summary>

        public int Diff(int a, int b)
        {
            return a - b;
        }
    }
    internal class Program
    {
        static void Main(string[] args)
        {
            SUB s = new SUB();
            Console.WriteLine("Sum :");
            Console.WriteLine(s.Sum(1,45,55));
            Console.WriteLine("Difference :");
            Console.WriteLine(s.Diff(22, 10));
            Console.ReadLine();
        }
    }
}
```

Output:



---

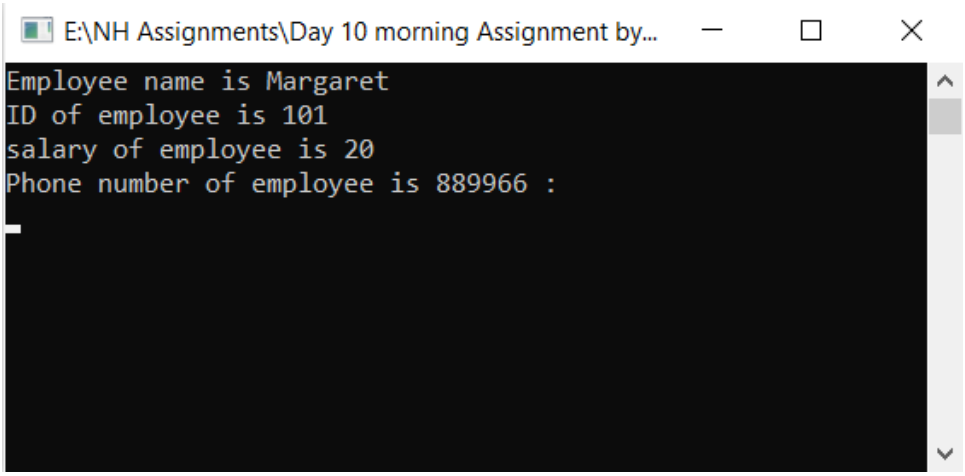
b. Code for Multi-Level Inheritance

```
using System;
using System.Collections.Generic;
```

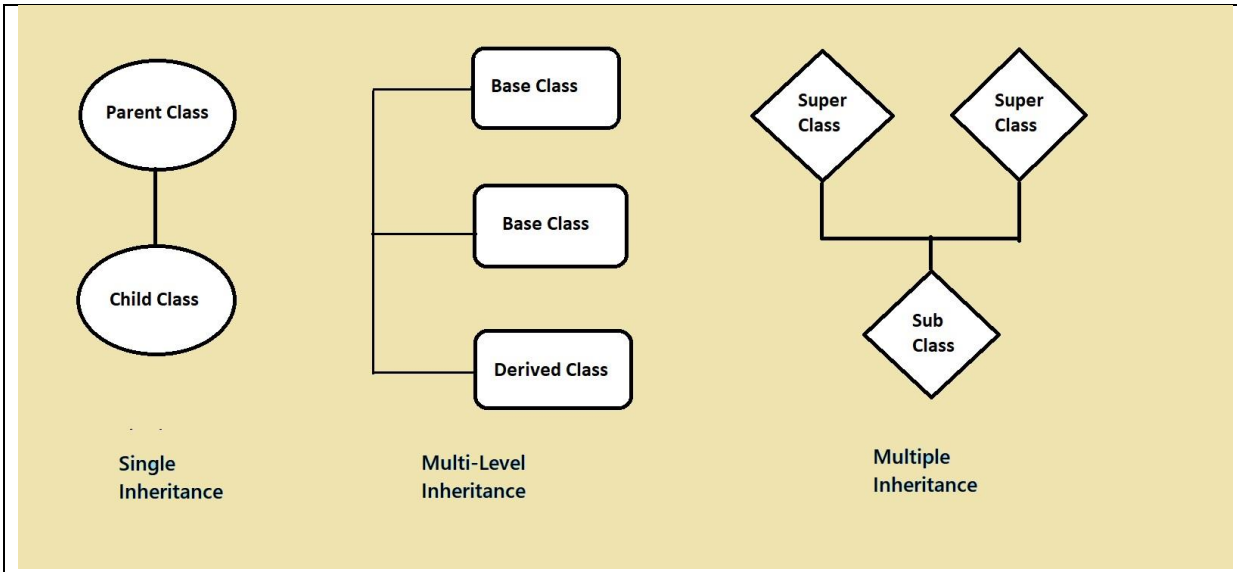
```
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace Day_10_Project_2
{
   //Author :Mary Margaret
   // Multi-Level Inheritance
   class Employee
   {
      /// <summary>
      /// This method gives name of employee
      /// </summary>
      public void Printname()
      {
         Console.WriteLine("Employee name is Margaret");
      }
      /// <summary>
      /// This method prints id of employee
      /// </summary>
      public void Printid(int a)
      {
         Console.WriteLine("ID of employee is {0}", a );
      }
   }
   class Salary : Employee
   {
      /// <summary>
      /// This method shows salary of employee
      /// </summary>
      public void Amount(int a)
      {
         Console.WriteLine("salary of employee is {0} ", a);
      }
   }
   class Phone : Salary
   {
      /// <summary>
      /// This method gives phone number of employee
      /// </summary>
      public void Phn(int a)
      {
         Console.WriteLine("Phone number of employee is {0} :" ,a);
      }
   }
   internal class Program
   {
      static void Main(string[] args)
      {
```

```
        Phone p = new Phone();
        p.Printname();
        p.Printid(101);
        p.Amount(20);
        p.Phn(889966);
        Console.ReadLine();
    }
  }
}
```

Output:



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



---

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

→ In Multiple inheritance, one class can have more than one superclass and inherit features from all its parent classes.

→ C# don't support multiple inheritance through classes because Multiple inheritances lead to ambiguity.

→ To implement multiple inheritances, we use Interfaces in C#.

5.What is polymorphism?

→ The ability of objects to perform different forms.

→ There are two types of Polymorphism:

1. Method overloading.
2. Method overriding.

→ When two methods have same name but with different parameters (Irrespective of return type) then it is known as “METHOD OVERLOADING”.

→ When two methods have same name, same parameters and same return type then it is known as "METHOD OVERRIDING".

---

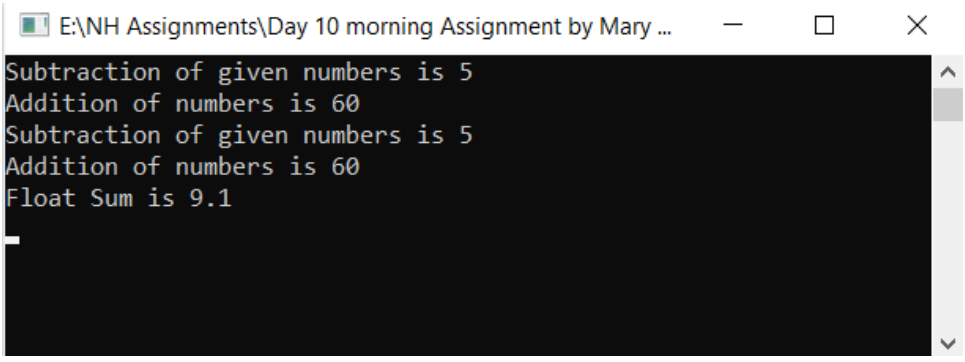
6. Write sample code for method overloading.

Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace Day_10_Project_3
{
   //Author:Mary Margaret
   //Purpose: Method Overloading
   class MethodOverloading
   {
      /// <summary>
      /// This method is to find sum
      /// </summary>
      public int Sub(int a, int b, int c)
      {
         return a - b - c;
      }

      public int Add(int a, int b, int c, int d)
      {
         return a + b + c + d;
      }
   }
   class Method_Overloading
   {
      /// <summary>
      /// This method is to find sum
      /// </summary>

      public int Sub(int a, int b, int c)
      {
         return a - b - c;
      }

      public int Add(int a, int b, int c, int d)
      {
```

```
        return a + b + c + d;
    }

    public float Add(float a, float b)
    {
        return a + b;
    }
  }
  internal class Program
  {
    static void Main(string[] args)
    {
        //Object creation for class1
        MethodOverloading m = new MethodOverloading();
        Console.WriteLine("Subtraction of given numbers is {0} ", m.Sub(8,2,1));
        Console.WriteLine("Addition of numbers is {0} ", m.Add(10,12,14,24));

        Method_Overloading m2 = new Method_Overloading();
        Console.WriteLine("Subtraction of given numbers is {0} ", m2.Sub(8, 2, 1));
        Console.WriteLine("Addition of numbers is {0} ", m2.Add(10, 12, 14, 24));
        Console.WriteLine("Float Sum is {0}", m2.Add(5.6f, 3.5f));
        Console.ReadLine();
    }
  }
}
```

Output:



---

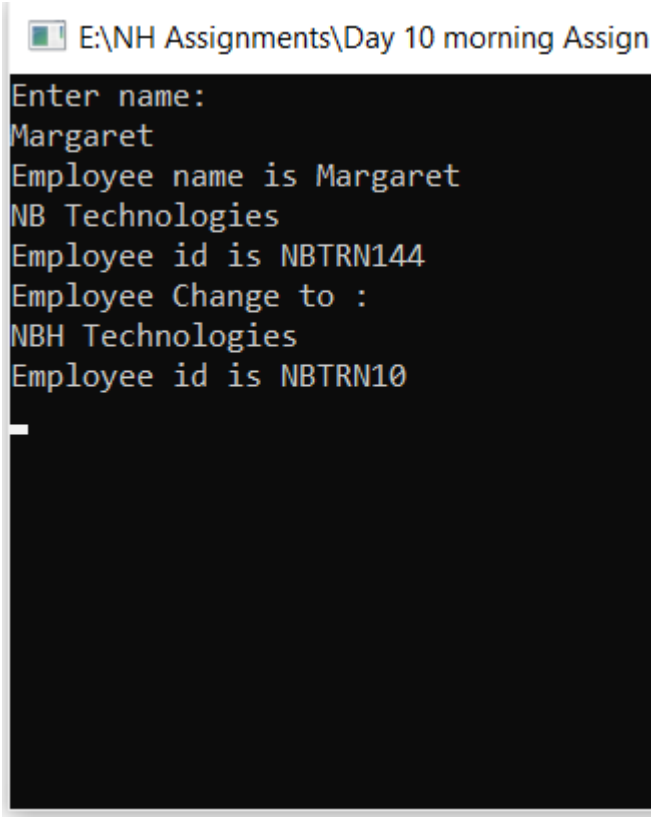
7. Write sample code for method overriding. [using new key word]

Code:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
namespace Day_10_Project_4
{
   //Author: Mary Margaret
   //Purpose: Method Overriding
   class A
   {
      string name;

      public void Reademp()
      {
         Console.WriteLine("Enter name:");
         name = Console.ReadLine();
      }

      public void Printemp()
      {
         Console.WriteLine("Employee name is {0}", name);
      }

      public void Company()
      {
         Console.WriteLine("NB Technologies");
      }
      public void ID()
      {
         string id = "NBTRN144";
         Console.WriteLine("Employee id is {0}", id);
      }
   }
   class B : A
   {

      public new void Company()
      {
         Console.WriteLine("NBH Technologies");
      }

      public new void ID()
      {
         string id = "NBTRN10";
         Console.WriteLine("Employee id is {0}", id);
      }
```

```
    }
    internal class Program
    {.I
        static void Main(string[] args)
        {

            A a = new A();
            a.Reademp();
            a.Printemp();
            a.Company();
            a.ID();

            B b = new B();
            Console.WriteLine("Employee Change to :");
            b.Company();
            b.ID();
            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_3
{
   //Author: Mary Margaret
   //code for method overriding using virtual, override keyword.
   public class Mobile
   {
      string name = "Samsung";
      //using virtual keyword
      public virtual void showdata()
      {
         Console.WriteLine("Name of mobile is " + name);
      }
   }

   class Model : Mobile
   {

      string s = "M30s";
      //using override keyword
      public override void showdata()
      {
         base.showdata();

         Console.WriteLine("Model is " + s);
      }
   }

   internal class Program
   {
      static void Main(string[] args)
      {
         Model m = new Model();
         m.showdata();
```

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

Output: