

i

DAY10 MORNING ASSIGNMENT
BY
ANDE MANOHAR
4TH FEB 2022

Q1. Write the two points discussed about inheritance in the class.

1. Inheritance is the process of Re-Using base class methods in the derived class.
2. Inheritance main goal is Re-usability and remove the duplicate code.

Q.2 Write the example code for

- a. single inheritance
- b. Multi inheritance

Code: Single inheritance

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

namespace DAY10_SINGLE_INHARITANCE
{
    class Algebra
    {
        /// <summary>
        /// This method is used for Addition
        /// </summary>
        /// <param name="a"></param>
        /// <param name="b"></param>
        /// <returns></returns>
        public int Add(int a, int b)
        {
            return a + b;
        }
    }

    class subtract : Algebra
    {
        /// <summary>
        /// This is used for subtraction
        /// </summary>
        /// <param name="a"></param>
        /// <param name="b"></param>
        /// <returns></returns>
        public int Sub(int a, int b)
        {

```

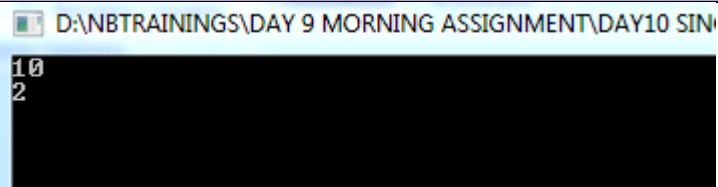
```

        return a - b;
    }

}

internal class Program
{
    static void Main(string[] args)
    {
        subtract obj = new subtract();
        Console.WriteLine(obj.Add(5, 5));
        Console.WriteLine(obj.Sub(4, 2));
        Console.ReadLine();
    }
}

```



OUTPUT:

Multi level inheritance:

```

Code: using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

namespace DAY10_multi_level_inharitance
{
    class Algebra
    {
        /// <summary>
        /// This method is used to find addition
        /// </summary>
        /// <param name="a"></param>
        /// <param name="b"></param>
        /// <returns></returns>
        public int Add(int a, int b )
        {
            return a + b;
        }
    }

    class Subtract :Algebra
    {
        /// <summary>
        /// This is used to find subtraction
        /// </summary>

```

```

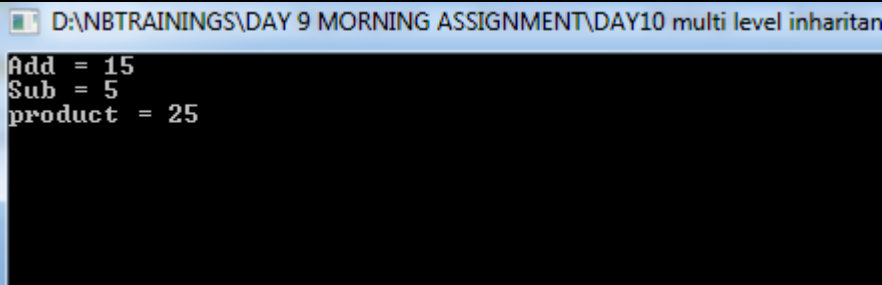
    /// <param name="a"></param>
    /// <param name="b"></param>
    /// <returns></returns>
    public int Sub(int a, int b)
    {
        return a - b;
    }
}
class product : Subtract
{
    /// <summary>
    /// his method is used for product
    /// </summary>
    /// <param name="a"></param>
    /// <param name="b"></param>
    /// <returns></returns>
    public int Multi(int a, int b)
    {
        return a * b;
    }
}

internal class Program
{
    static void Main(string[] args)
    {
        product obj = new product();
        Console.WriteLine($"Add = {obj.Add(10, 5)}");
        Console.WriteLine($"Sub = {obj.Sub(10, 5)}");
        Console.WriteLine($"product = {obj.Multi(5, 5)}");

        Console.ReadLine();
    }
}

```

Output:

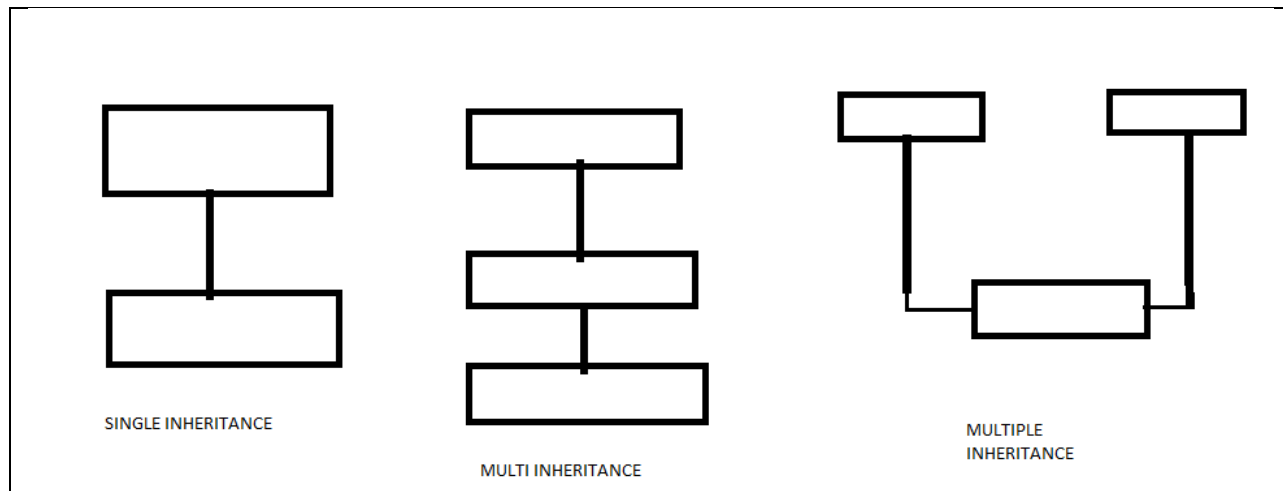


```

D:\NBTRAININGS\DAY 9 MORNING ASSIGNMENT\DAY10 multi level inharitan
Add = 15
Sub = 5
product = 25

```

Q3. Pictorial representation of 3 types of inheritances



Q4. Write why multiple inheritance is not supported in c#

In C# compiler is designed not to support multiple inheritance because it causes ambiguity of methods from different base class.

This is caused by diamond shape problems of two classes if two classes B and C inherit from A, and class D inherits from both B and C so, multiple inheritance is not possible in c#

Q5. What is polymorphism?

Polymorphism is the ability of an object which acts as like many forms.

Two types of polymorphism

- Method overloading
- Method overriding

Q6. Write simple code for method overloading

Coad:

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

namespace Method_overloading
{
    class Algebra
    {
        /// <summary>
        /// This is for addition
        /// </summary>
        /// <param name="a"> </param>
        /// <param name="b"> </param>
        /// <returns> </returns>
        public int Add(int a, int b)
```

```

    {
        return a + b;
    }
    /// <summary>
    /// This is used for addition
    /// </summary>
    /// <param name="a"></param>
    /// <param name="b"></param>
    /// <param name="c"></param>
    /// <returns></returns>
    public int Add(int a, int b, int c)
    {
        return a + b + c;
    }
    /// <summary>
    /// This is used for addition
    /// </summary>
    /// <param name="a"></param>
    /// <param name="b"></param>
    /// <param name="c"></param>
    /// <param name="d"></param>
    /// <returns></returns>
    public int Add(int a, int b, int c, int d)
    {
        return a + b + c + d;
    }
}
internal class Program
{
    static void Main(string[] args)
    {
        Algebra m = new Algebra();
        Console.WriteLine("sum1 is {0}", m.Add(5, 2));
        Console.WriteLine("sum2 is {0}", m.Add(3,5,2));
        Console.WriteLine("sum3 is {0}", m.Add(2,5,1,2));

        Console.ReadLine();
    }
}

```

Output:

```
D:\NBTRAININGS\DAY 9 MORNING ASSIGNMENT\Method overloading\Method overlo...
sum1 is 7
sum2 is 10
sum3 is 10
```

Q7. Write sample code for method overriding

Code:

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

namespace DAY10_method_overriding
{
    class English
    {
        /// <summary>
        /// This Metthod is to print
        /// </summary>
        public void Print()
        {
            Console.WriteLine("good morning");
        }

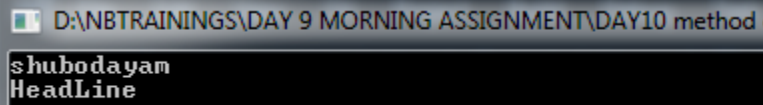
        public void PrintHeadLine()
        {
            Console.WriteLine("HeadLine");
        }
    }

    class Telugu : English
    {
        public new void Print()
        {
            Console.WriteLine("shubodayam");
        }
    }

    internal class Program
    {
```

```
static void Main(string[] args)
{
    Telugu obj = new Telugu();
    obj.Print();
    obj.PrintHeadLine();
    Console.ReadLine();
}
```

Output:



D:\NBTRAININGS\DAY 9 MORNING ASSIGNMENT\DAY10 method

shubodayam
HeadLine