

Method Overloading (function Overloading) ①

Multiple methods of same name with different parameters in a single program or class is called method overloading.

Ex - namespace ~~Ex~~ object [Find the area]
{
 class overload
 {

 static double Area (double l, double w)

 {
 return (l * w);
 }

 static double Area (double R)

 {
 const double PI = 3.14;
 return PI * R * R;
 }

 static void main ()

 {
 Console.WriteLine (Area (10.2, 3.25));

 Console.WriteLine (Area (20));

 Console.ReadLine ();

 }

 }

Method Overriding

(2)

Method overriding mean same name method used in different classes.

Ex namespace Mycel-

{
class new

{

class A

{ public virtual void show().

{ Console.WriteLine("Hello friend");
}

}

class B : A

{ public virtual void show()

{ Console.WriteLine("Hello my lovely friend");
} }

class C : A

{ public override void show().

{ Console.WriteLine("I am your new friend");
}

}

public static void main ()

(2)

```
{  
    A obj = new A();  
    A obj; ← Assign the object  
    obj = new B();  
    obj.show();  
    obj = new C();  
    obj.show();  
}
```

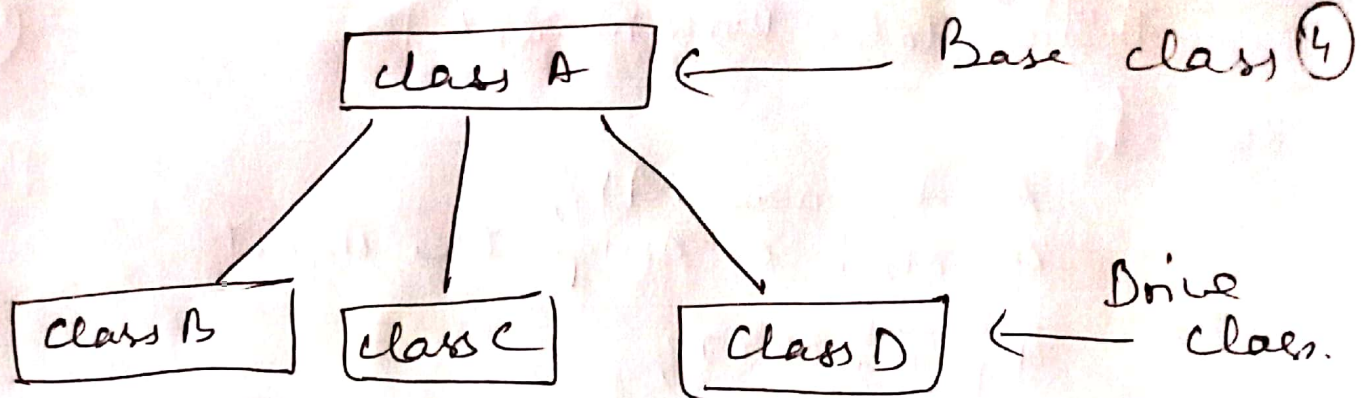
Q1 - Hello my lovely friend.
I am your new friend.

INHERITANCE

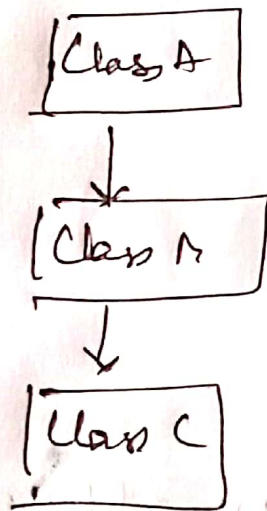
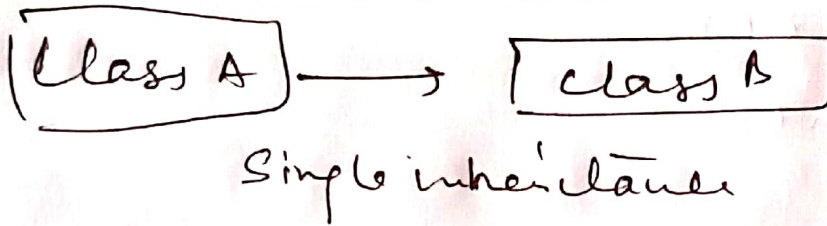
Inheritance is a mechanism of sharing the members among the classes.

- First create class → Base class
- Other create class → Derive class.

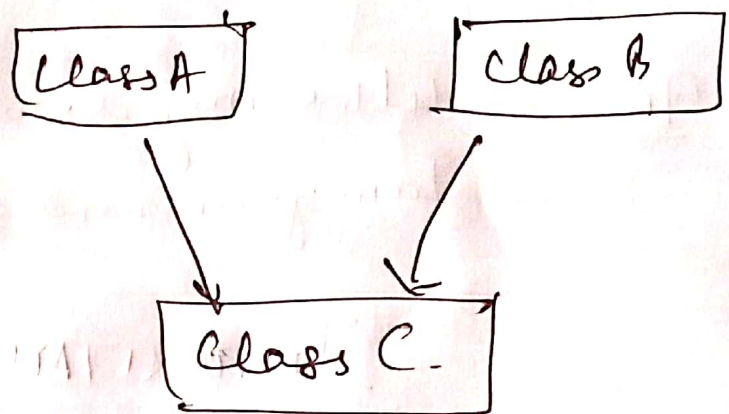
(Derive classes inherit the properties of Base class) Base class directly not take part in program.



Hierarchical inheritance



Multilevel.



~~Multiple inheritance.~~

↑
Not exist in OOP concept

syntax :- for Drive class

class driven-class-name : base-class-name.

{ member of class .
}

Single Inheritance

5

namespace Inherit-

```
{
    class A
    {
        public void first()
        {
            Console.WriteLine("Welcome boys be  
prepared for C++ Lang");
        }
    }
}
```

class B : A // class B is derived class.

```
{
    public void second()
    {
        Console.WriteLine("What up guys why  
r u so lazy");
    }
}
```

class Simple

```
{
    public static void main (String[] args)
    {
        class B obj = new B();
        B obj = new B();
        obj.first();
        obj.second();
    }
}
```

SP:
Welcome boys be
prepared for C++
What up guys why
r u so lazy

Multilevel Inheritance

(4)

```
class A
{
    public void welcomeFirst()
    {
        Console.WriteLine("Welcome to programming Language");
    }
}

class B : A
{
    public void Second()
    {
        Console.WriteLine("Plz seriously doing Program");
    }
}

class C : B
{
    public void Third()
    {
        Console.WriteLine("I think you're not interesting");
    }
}

class Simple
{
    static void main (String[] args)
    {
        C obj = new C();
        obj.First();
        obj.Second();
        obj.Third();
    }
}
```

Op
Welcome to prog. Lang :
Plz seriously doing Program
I think you are not interesting.

Hierarchical Inheritance

class A

{ public void first ()

{ console.WriteLine ("Very careful in programming");
}

class B : A

{ public void second ()

{ console.WriteLine ("Nothing is impossible");
}

class C : A

{ public void third ()

{ console.WriteLine ("No need must try
the program up level");
}

} class Simple

{ static void main (String [] args)

{ B obj = new B();

C K = new C();

obj.first();

obj.second();

K.third(); } }