

## Object Oriented Strategy : Inheritance

There are four Object Oriented Paradigms of C# as

1. Encapsulation
2. Abstraction
3. Polymorphism
4. Inheritance

Inheritance means reusability.

By using the concept of one class can reuse the characteristics and behaviour of another class.

There are two types of inheritance as

1. Single level inheritance
2. Multilevel inheritance

Multiple inheritance is not allowed in C#.

### Application 1 :

- Application which demonstrates concept of single level inheritance.
- In this application we design one class named as base which contains one characteristic and one behaviour.
- Derived class inherited the base class which contains its own characteristics and behaviour.

using System;

class Base

```
{  
    public int i;           // Characteristics  
  
    public void fun()      // Behaviour  
    {  
        Console.WriteLine("Inside Base fun");  
    }  
}
```

class Derived : Base // Derived class inherited the base class

```
{  
    public int j;           // Characteristics  
  
    public void gun()      // Behaviour  
    {  
        Console.WriteLine("Inside Derived gun");  
    }  
}
```

class Program

```
{
```

```
static void Main(string[] args)
{
    Base bobj = new Base();

    Derived dobj = new Derived();

    bobj.fun();

    dobj.fun();        // Access method which is inherited from base class
    dobj.gun();
}
}
```

---

## Application 2 :

Application which demonstrates concept multilevel inheritance.

using System;

```
class Base
{
    public int i;
    public void fun()
    {
        Console.WriteLine("Inside Base fun");
    }
}

class Derived : Base
{
    public int j;
    public void gun()
    {
        Console.WriteLine("Inside Derived gun");
    }
}

class MyDerived : Derived
{
    public int k;
    public void sun()
    {
        Console.WriteLine("Inside MyDerived sun");
    }
}

class Program
{
```

```
static void Main(string[] args)
{
    Base bobj = new Base();
    Derived dobj = new Derived();
    MyDerived mobj = new MyDerived();

    bobj.fun();
    dobj.fun();
    dobj.gun();
}
}
```

