C# applications for Method Overriding

Polymorphism means Single name multiple behaviour.

Thera are two types of polymorphisms as

1. Compile time polymorphism
2. Run time polymorphism

We can achieve run time polymorphism using function overriding.

In C# we can use virtual and override keywords for overriding.

To get the effect of overriding there should be atlas single level inheritance and upcasting.

Application 1 :

```
using System;

    class Base
    {
        public void fun()
        {
            Console.WriteLine("In base fun");
        }

        // Virtual method of class
        public virtual void gun()
        {
            Console.WriteLine("In base gun");
        }
    }

    // Inherit Base class
    class Derived : Base
    {
        // Redefinition of fun method
        public void fun()
        {
            Console.WriteLine("In Derived fun");
        }
```

```
    // Overriding gun method
     public override void gun()
    {
        Console.WriteLine("In Derived gun");
    }
}

class Program
{
    static void Main(string[] args)
    {
        Base bobj1 = new Base();

        Derived dobj = new Derived();

        bobj1.fun();
        bobj1.gun();

        dobj.fun();
        dobj.gun();

        // Upcasting
        Base bobj2 = new Derived();
        bobj2.fun();
        bobj2.gun();
    }
}
```