**Overloading**

Overloading with same signature and **ref** parameter is possible, but not possible with **out** parameter.

Ex In this third statement is invalid.

public void Print(int i){}

public void Print(ref int i){}

public void Print(out int i){i = 0;} // Will not works.

**Overriding**

class Base

{

public virtual void Show()

{

Console.WriteLine("Base");

}

}

class Derived : Base

{

public override void Show()

{

Console.WriteLine("Derived");

}

}

Case 1

Derived obj = new Derived();

obj.Show();

OP :

Derived

Case 2

Base obj = new Derived();

obj.Show();

OP :

Derived

**Method Hiding / Showing**

class Parent

{

public void Show()

{

Console.WriteLine("Parent");

}

}

class Child : Parent

{

public new void Show()

{

Console.WriteLine("Child");

}

}

Case 1

Child obj = new Child();

obj.Show();

OP

Child

Case 2

Parent obj = new Child();

obj.Show();

OP

Parent