

## C#中隐藏(new)和方法重载(override)区别

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

```
namespace NewOverride
```

```
{
    class Program
    {
        /*
```

重写和隐藏的定义:

重写: 基类方法声明为virtual(虚方法), 派生类中使用override申明此方法的重写.

隐藏: 基类方法不做申明 (默认为非虚方法), 在派生类中使用new声明此方法的隐藏。

自己的理解:

比如父类A,有个方法标记为virtual,a(){}子类B继承A,也声明一个方法a(){}

如果B里面的a()使用override,那么访问A的方法时实际上调用了B里面声明的方法,相当于A的方法被覆盖了,new就不是,

访问A的a还是A里面定义的方法,访问B就是B里面定义的方法.

说白了: new是覆盖, override是重载, “覆盖”并不意味着“删除”, 但“重载”意味着“删除”,

这就是“覆盖”和“重载”的区别

隐藏(new)示例:

```
*/
```

```
static void Main(string[] args)
```

```
{
```

```
    B b = new B();
```

```
    b.ClassA();
```

```
    A a = b;
```

```
    a.ClassA();
```

```
    Console.WriteLine("\n");
```

```
    B2 b2 = new B2();
```

```
    b2.ClassA2();
```

```
    A2 a2 = b2;
```

```
    a2.ClassA2();
```

```
    Console.ReadKey();
```

```
}
```

```
}
```

```
//
```

```
class A
```

```
{
```

```
    public void ClassA()
```

```
    {
```

```
        Console.WriteLine("A.ClassA()");
```

```
    }
```

```
}
```

```
class B : A
```

```
{
```

```

    new public void ClassA()
    {
        Console.WriteLine("B.ClassA()");
    }
}

//
class A2
{
    public virtual void ClassA2()
    {
        Console.WriteLine("A2.ClassA2()");
    }
}
class B2 : A2
{
    public override void ClassA2()
    {
        Console.WriteLine("B2.ClassA2()");
    }
}
}

```

```

File: //E:/Deam/C#//B27
B.ClassA()
A.ClassA()

B2.ClassA2()
B2.ClassA2()
-

```