昨天有个网友问我继承里面的new和override关键词有啥区别，呃，我们来看个例子就知道了

new ==》隐藏父类同名方法  override==》覆盖

定义一个父类：

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | public class People  {      /// <summary>      ///ushort参考： <https://msdn.microsoft.com/zh-CN/library/cbf1574z.aspx>      /// </summary>      public string Name { get; set; }      public ushort Age { get; set; }        public People(string name, ushort age)      {          this.Name = name;          this.Age = age;      }      public void Hi()//People      {          Console.WriteLine("Name: " + this.Name + " Age: " + this.Age);      }      public virtual void Show()//People      {          Console.WriteLine("Name: " + this.Name + " Age: " + this.Age);      }  } |

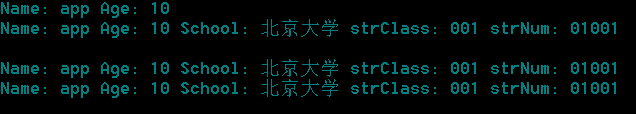
　　定义一个子类:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50 | public class Student : People  {      #region 属性      /// <summary>      /// 学校      /// </summary>      public string School { get; set; }      /// <summary>      /// 班级      /// </summary>      public string StrClass { get; set; }      /// <summary>      /// 学号      /// </summary>      public string StrNum { get; set; }      #endregion        #region 构造函数      /// <summary>      /// 调用父类构造函数      /// </summary>      /// <param name="name"></param>      /// <param name="age"></param>      public Student(string name, ushort age) : base(name, age)      {        }      public Student(string name, ushort age, string school, string strClass, string strNum) : this(name, age)      {          this.School = school;          this.StrClass = strClass;          this.StrNum = strNum;      }      #endregion        /// <summary>      /// new-隐藏      /// </summary>      public new void Hi()//Student      {          Console.WriteLine("Name: " + this.Name + " Age: " + this.Age + " School: " + this.School + " strClass: " + this.StrClass + " strNum: " + this.StrNum);      }      /// <summary>      /// override-覆盖      /// </summary>      public override void Show()//Student      {          Console.WriteLine("Name: " + this.Name + " Age: " + this.Age + " School: " + this.School + " strClass: " + this.StrClass + " strNum: " + this.StrNum);      }  } |

　　调用一下

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | static void Main(string[] args)          {              People p = new Student("app", 10, "北京大学", "001", "01001");              p.Hi(); p.Show();                System.Console.WriteLine();                Student s = p as Student;              s.Hi(); s.Show();                System.Console.ReadKey();          } |

　　输出：



**可以发现，用new修饰的方法，其父方法并没有被改变，只是被隐藏了**

**用override重写的方法却被覆盖，从而改变了（多态）**