



## 二、基本控件

### 1. 标签 A Label

作用：显示信息(不能输入信息)

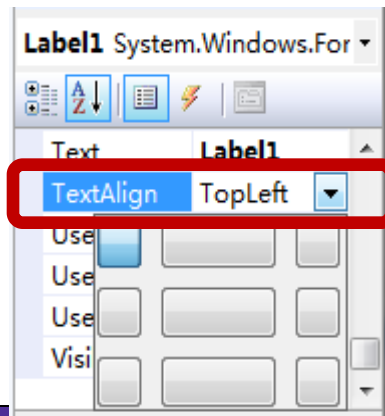
#### (1) 主要属性

➤ **Text、TextAlign:**

➤ **Image、ImageAlign**

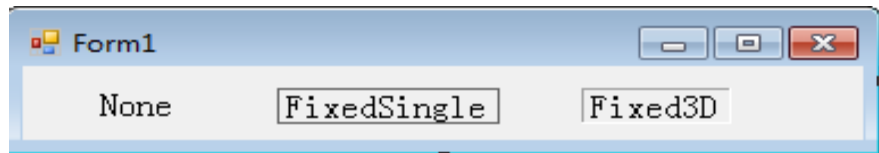
设置文本、图案及  
对齐方式

对齐方式是枚举  
类型，有9种



## ➤ **BorderStyle** --边框样式

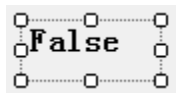
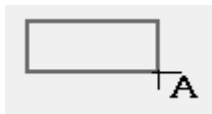
**None** (默认值)、**FixedSingle**、**Fixed3D**



## ➤ **AutoSize** --否能自动调整大小

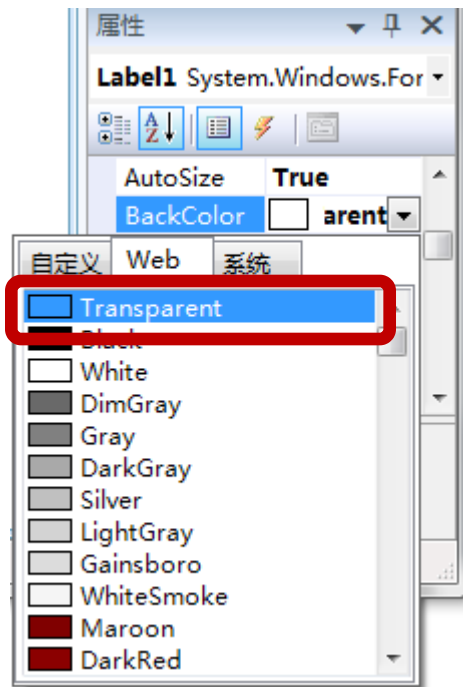
默认为**True**，框与显示的文字同大；

**False**，框大小不变，若正文超长则截断显示。





# 如何防止标签遮盖窗体背景图片？



➤ **BackColor**-- 背景色  
默认与窗体背景相同

设置**Color.Transparent**值。





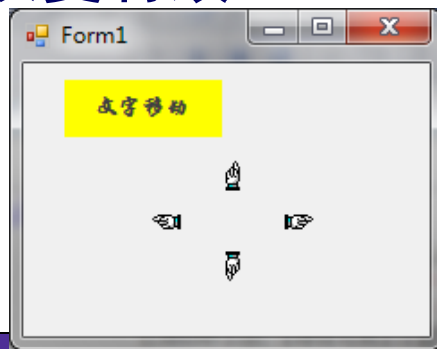
## 例2.3 手指图标的指向, 实现文字移动。

要求:

- ◆4个标签显示4个方向的“手指”图标;
- ◆单击某方向, 文字向该方向移动15个像素;
- ◆超出窗体边界, 提示并使该标签无效;  
当反向的手指移动时, 该标签恢复有效。

手指图标文件名为

Point02.ico~Point05.ico,  
在安装的vs系统文件夹下。





Label1.Top

Label1.Top+  
Label1.Height

Width

文字移动

Height

0,0

文字移动

文字移动

移动

窗体

Height

Label1.Left

Label1.Left+ Label1.Width

◆ 超左边界

$\text{Label1.Left} + \text{Label1.Width} < 0$

◆ 超上边界

$\text{Label1.Top} + \text{Label1.Height} < 0$

- ◆左移Label3事件过程, 左移15像素;
- ◆超出窗体, 提示该标签无效;



```
Sub Label3_Click(...) Handles Label3.Click
```

```
    Label5.Left = Label5.Left - 15    '左移
```

```
    Label4.Enabled = True
```

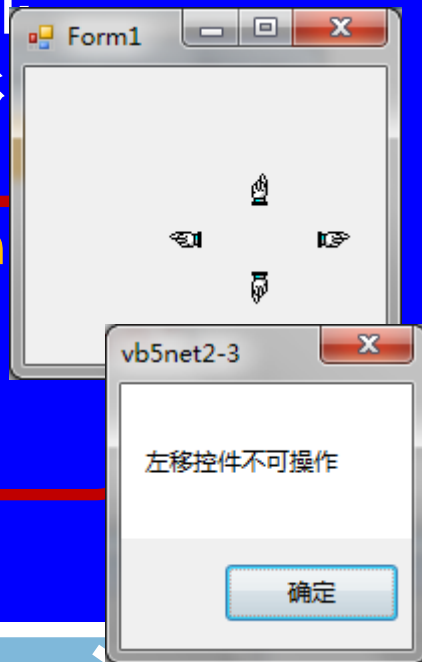
```
    If Label5.Left + Label5.Width < 0 Then
```

```
        Label3.Enabled = False
```

```
        MsgBox("左移控件不可操作")
```

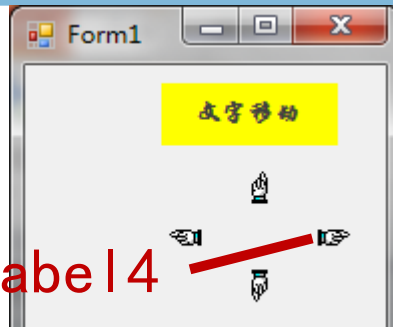
```
    End If
```

```
End Sub
```





- ◆左移Label3事件过程, 左移15像素;
- ◆超出窗体, 提示该标签无效;
- ◆标签何时有效? 反向操作右移按钮Label4



```
Sub Label3_Click(...) Handles Label3.Click
```

```
Label5.Left = Label5.Left - 15 '左移
```

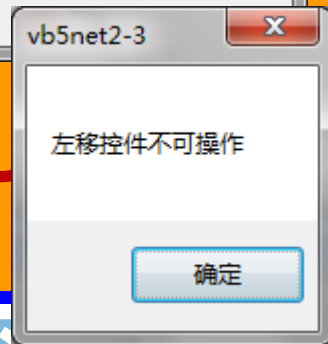
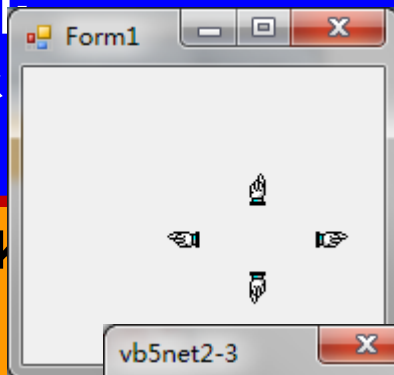
```
Label4.Enabled = True
```

```
Sub Label4_Click(...) Handles Label4.Click
```

```
...
```

```
Label3.Enabled = True
```

```
'右移
```



```
End Sub
```





## 2. 命令按钮



作用：程序运行时执行相应的事件

(1) 主要属性

➤ **Text**、**TextAlign**

➤ **Image**、**ImageAlign**

设置文本、背景图案及对齐方式

(2) 主要事件

一般接收**Click**事件







### 3. 图片框控件 PictureBox

作用：显示图片

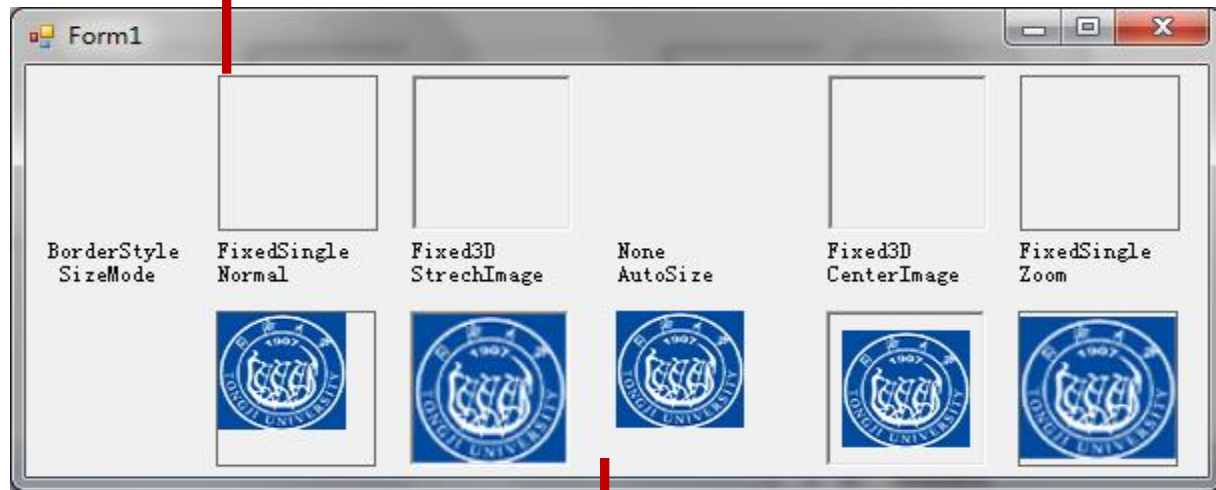
(1) 主要属性

- **Image:** 装入图片
- **BorderStyle:** 设置图片边框样式
- **SizeMode:** 图片框与图片显示模式





## BorderStyle属性设置效果



## SizeMode属性设置效果

StretchImage属性值图片随图片框大小而变

AutoSize属性值框随着图大小而变





## 实验1.5 图片缩放

实现方法：根据**SizeMode**的**StretchImage**属性特点，实质是对框的缩放

初始：将图片框大小存放于模块级变量，供缩放引用

```
Dim h%, w%
```

```
Sub Form1_Load(...) Handles
```

```
w = PictureBox1.Width
```

```
h = PictureBox1.Height
```

```
End Sub
```





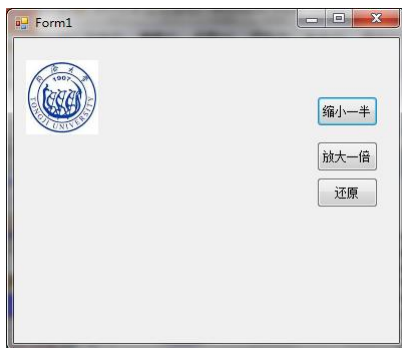
缩小一半：

Sub **Button1\_Click(...)** Handles Button1.Click

```
PictureBox1.Width = w / 2  
PictureBox1.Height = h / 2
```

将原图片框尺寸缩小一半  
赋值给图片框

End Sub



放大一倍、还原功能  
如何方便地实现？

拓展：如何实现对图片连续缩放？





## 4. 文本框

abl TextBox

作用：输入、显示、编辑文本内容。

(1) 主要属性：

- **Text** 显示或输入的正文内容
- **PasswordChar** : 设置输入密码显示的字符，  
如 “\*”，输入内容在Text属性
- **MaxLength**: 可输入字符的最大长度。
- **TextLength** : 获取输入的字符长度。

在VB.NET中一个西文字符与一个汉字都是一个字





## ➤ MultiLine 多行属性

**False**, 设置为**True**时, 可输入或显示多行正文;

## ➤ ScrollBars 滚动条属性

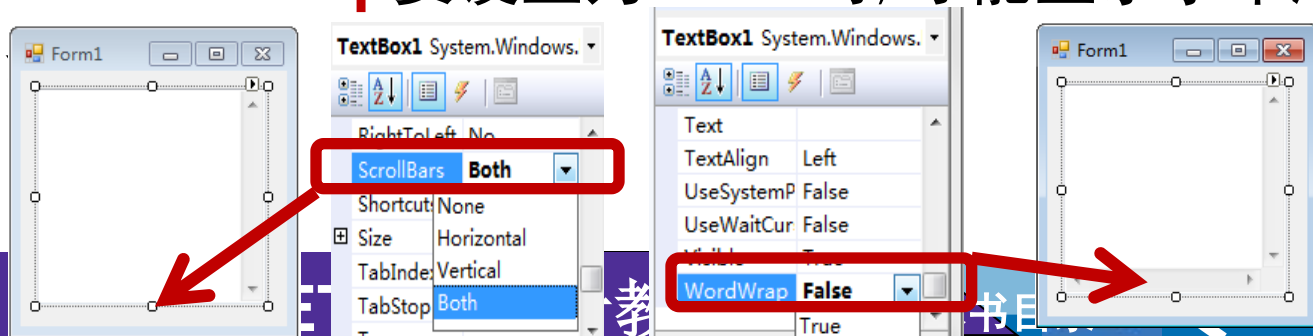
**None**、Horizontal、Vertical、Both

ScrollBars设置为非None值为什么无效?

**MultiLine 为False**

ScrollBars为Both或Horizontal, 但无水平滚动条?

## ➤ WordWrap要设置为**False**时, 才能显示水平滚动





第1个值为0

## ➤ 选定文本内容

**SelectionStart**

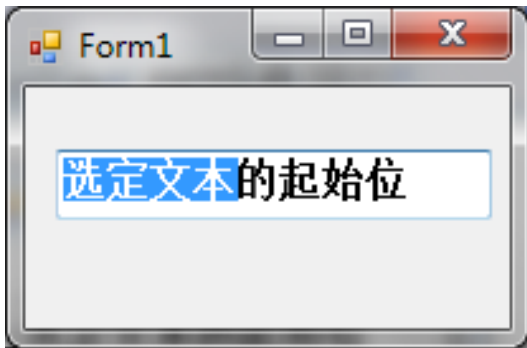
选定的正文开始位置

**SelectionLength**

选定的正文长度

**SelectedText**

选定的正文内容



**TextBox1.SelectionStart = 0**

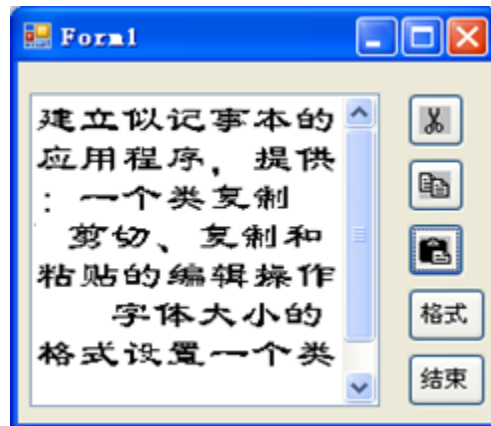
**TextBox1.SelectionLength = 4**

**TextBox1.SelectedText = "选定文本"**



例2.6建立一个类似记事本的简单程序。要求：

- ◆ 具有对文本编辑操作
- ◆ 具有字体大小的格式设置



分析：

- ◆ 建立一个有滚动条的文本框输入文本。
- ◆ 利用文本框的“选定”3属性实现对选定文本的“剪切、复制和粘贴”的编辑操作；
- ◆ “格式”设置利用**Font**对象实现。







## (2) 主要事件

### ➤ **TextChanged** 事件

当**Text**属性值发生改变时引发该事件

**例**在文本框每输入一个字符就激发一次

### ➤ **KeyPress** 事件

按下并且释放键盘上的一个键时，引发焦点所在控件的KeyPress事件，所按键的值存放在参数**e对象的KeyChar**中

对回车的判断  $\text{Asc}(e.\text{KeyChar})=13$

**举例：比较两个事件区别**



## ➤ **LostFocus**事件

对象失去焦点时发生**LostFocus**事件

作用：对输入的数据进行验证

## ➤ **GotFocus**事件

与**LostFocus**事件相反，对象获得焦点时发生





## (3) 方法

### ➤ **Focus**方法

作用：把光标移到指定的对象中。

形式：

**[对象.] Focus()**

该方法还可以用于其他控件如：

CheckBox、ListBox、ComboBox等





例2.5 简易加法器。要求对输入的数据进行合法性检验，过滤不合法的数据。

Form1

12 + 34 = 46

分析：

- **TextBox1**输入结束按**Tab**键进行合法性检验
- **TextBox2**输入结束按**回车键**进行合法性检验
- 合法性检验要求输入的数字数据，调用函数 **IsNumeric(s)**
- 非法数据时处理：清除文本框内容焦点回到文本框

**LostFocus**事件

**KeyPress**事件





**TextBox1**输入结束按**Tab**键进行合法性检验

```
Sub TextBox1 LostFocus(...)
```

```
  If Not IsNumeric(TextBox1.Text) Then
```

- ◆输入数据合法否？
- ◆非法数据时的处理

```
  End If
```

```
End Sub
```

134

1c4

IsNumeric(TextBox1.Text)

True

False





**TextBox2**输入结束按**回车键**进行合法性检验

```
Sub TextBox2_KeyPress (..., e )
```

```
    If Asc(e.KeyChar) = 13 Then
```

```
        If Not IsNumeric(TextBox2.Text) Then
```

```
            TextBox2.Text = ""
```

```
            TextBox2.Focus()
```

```
        End If
```

思考：在文本框输入结束按回车、回车键均  
进行数据合法性检验？





## TextBox3 获得焦点进行计算

```
Sub TextBox3_GotFocus(...)  
    TextBox3.Text = Val(TextBox1.Text) + _  
                    Val(TextBox2.Text)  
End Sub
```

