

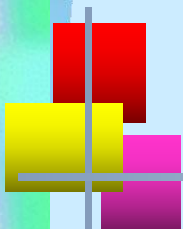
# 第9章 控 件

## 知识要点

- 控件的分类
- 常用内部控件的功能、用法及使用技巧
- 控件数组的功能、用法及使用技巧

## 学习任务

- 了解控件的分类、常用内部控件及控件数组的功能
- 理解控件的各种属性、方法和事件的功能及含义
- 掌握常用内部控件的使用方法和技巧及控件数组的创建方法



控件实际上就是一些包含在窗体中的对象，一般用于在应用程序中输入和显示文本，或者访问其他的应用程序及数据。

### 在设计时，将某一控件添加到窗体中的方法：

- ✦ 双击工具箱中对应的控件图标
- ✦ 单击工具箱中对应的图标，窗体中的鼠标指针变成“十”字形，在窗体中拖动鼠标，当产生的虚线框大小合适后松开鼠标左键

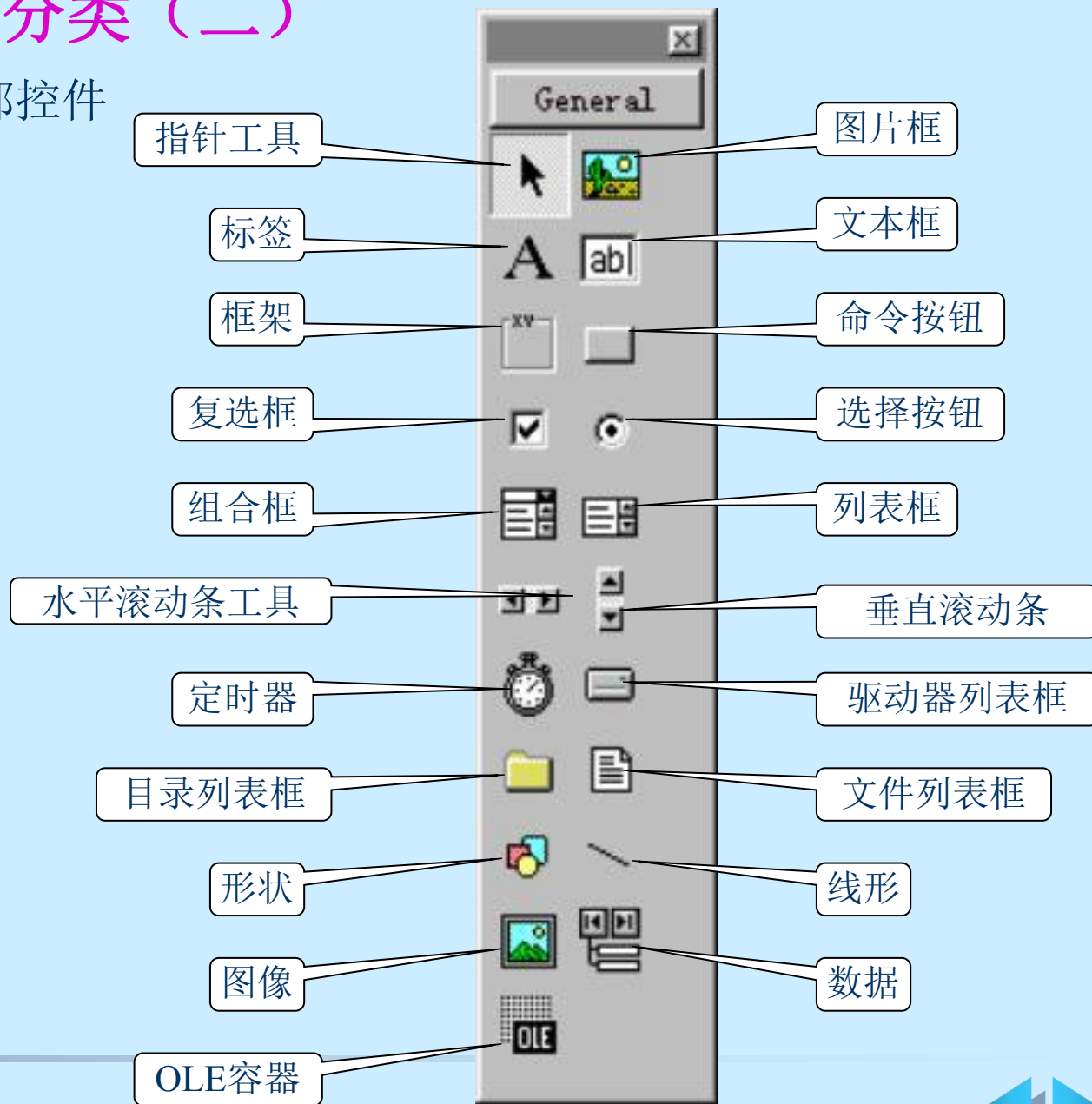
## 9.1 控件的分类（一）

- ✦ **内部控件：**由Visual Basic提供的控件，包含在Visual Basic的扩展名为.exe的文件中。显示在工具箱中，不能被删除
- ✦ **ActiveX控件：**是一些扩展名为.ocx的独立文件
- ✦ **可插入对象：**是一些能够添加到工具箱中并可以作为控件使用的对象



# 9.1 控件的分类 (二)

工具箱中的内部控件



## 9.2 关于控件的几点说明（一）

### 控件的名字（Name）属性

为控件命名遵循的原则：

- ✦ 用前缀描述控件所属的类，其后为控件的描述性名字
- ✦ 只能包含字母、数字和下划线字符（\_），不允许有标点符号字符和空格，
- ✦ 长度不能超过40个字符

### 设置控件的位置和大小

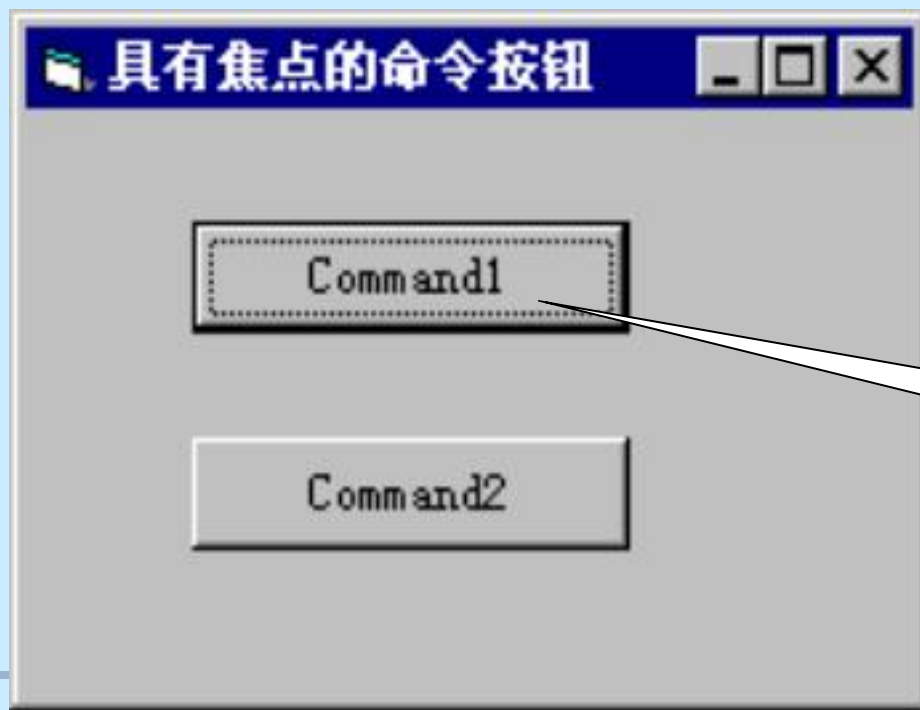
- ✦ **Left和Top属性**：用来设置控件左边缘或上边缘与窗体左边缘或上边缘的相对距离
- ✦ **Width和Height属性**：用来设置控件的宽度和高度
- ✦ **Move方法**：用来移动窗体或者控件  
(语法: `[object.]Move left[, top[, width[, height]]]`)



## 9.2 关于控件的几点说明（二）

### 设置对象的焦点（Focus）

- ✦ 焦点表示对象接收键盘输入的能力
- ✦ 将焦点赋给某一对象有以下两种方法：
  - 运行时选择某一对象
  - 在代码中调用SetFocus方法



此命令按钮  
具有焦点



## 9.2 关于控件的几点说明（三）

### 设置Tab键的顺序

- ✦ Tab键的顺序就是当按下Tab键时，焦点在窗体中的各控件间移动的顺序
- ✦ 在缺省情况下，Tab键的顺序与控件的建立顺序相同
- ✦ 改变Tab键的顺序，需要设置控件的TabIndex属性值
- ✦ 从Tab键顺序中删除控件，只要将其TabStop属性设为False即可



## 9.3 一般类控件（一）

### 命令按钮（CommandButton）（一）

✦ **基本功能：** 单击命令按钮，系统将执行相应的程序，完成一定的任务

✦ **常用属性**

**Caption属性：** 设置或返回命令按钮上显示的内容

（语法： `Command1.Caption [=string]`）

**Enabled属性：** 确定命令按钮控件是否能够响应事件

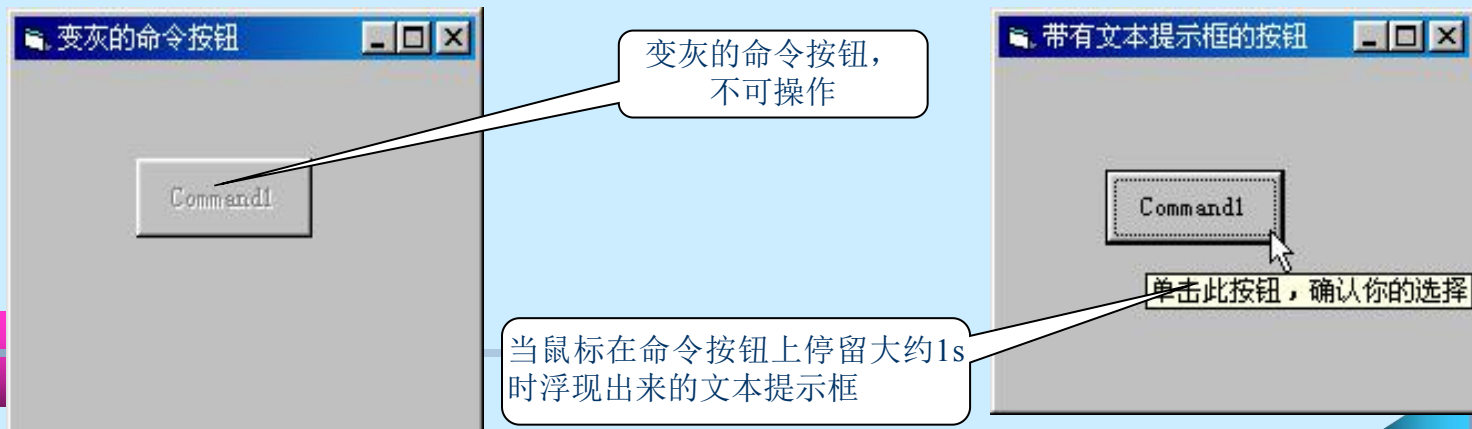
（语法： `Command1.Enabled [=boolean]`）

**Visible属性：** 确定对象在窗体中是否可见

（语法： `Command1.Visible [=boolean]`）

**ToolTipText属性：** 返回或设置当鼠标在命令按钮上停留时，在该命令按钮下面的黄色小方框中显示的提示文本

（语法： `Command1.ToolTipText [=string]`）





## 9.3 一般类控件（二）

### 命令按钮（CommandButton）（二）

✦ **常用方法：** Drag、Move、SetFocus等多种方法，Setfocus方法最常用

✦ **常用事件过程**

**Click单击事件：** 在命令按钮上单击鼠标左键，或当命令按钮具有焦点时，按下空格（Spacebar）键或按下命令按钮的快捷键

（语法：Private Sub Command1\_Click（））

**Gotfocus事件：** 当命令按钮获得焦点时触发

（语法：Private Sub Command1\_GotFocus（））

**LostFocus：** 当命令按钮失去焦点时触发

（语法：Private Sub Command1\_LostFocus（））





## 9.3 一般类控件（三）

### 标签（Label）（一）

✦ **基本功能：** 用来在运行时显示一些只读的文本信息

#### ✦ 常用属性（一）

**Caption属性：** 设置或返回显示在标签中的内容

（语法： `Label1.Caption [=string]`）

**Alignment属性：** 返回或设置标签中文本的对齐方式

（语法： `Label1.Alignment [=number]`）

**AutoSize属性：** 决定标签控件是否能够根据输入的文本的长度自动改变尺寸，以适应其内容的要求

（语法： `Label1.AutoSize [=boolean]`）

**WordWrap属性：** 决定标签是否能够根据输入的文本的高度自动改变尺寸，以适应其内容的要求

（语法： `Label1.WordWrap [=boolean]`）



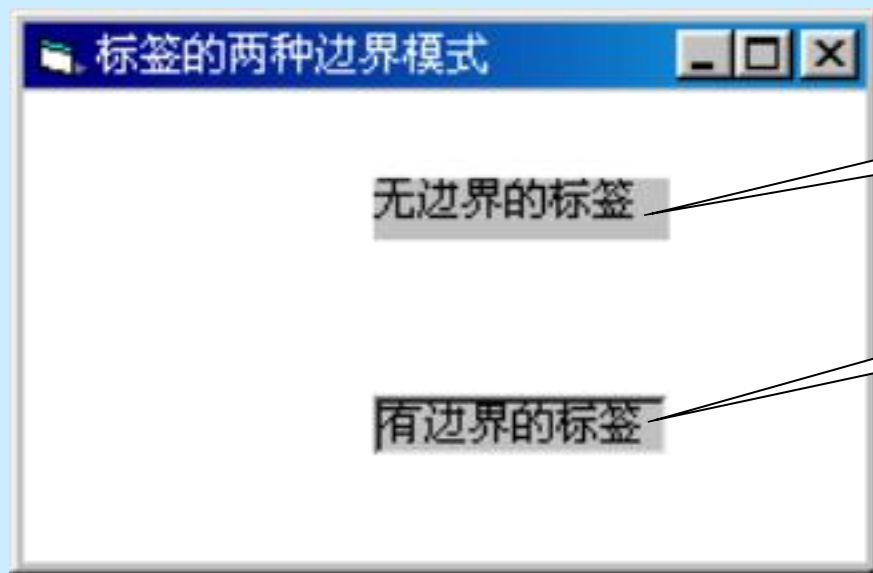
## 9.3 一般类控件（四）

### 标签（Label）（二）

#### 常用属性（二）

**BackColor属性：**返回或设置标签的背景模式  
(语法: `Label1.BackColor [=number]`)

**BorderStyle属性：**返回或设置标签的边界模式  
(语法: `Label1.BorderStyle [=number]`)



无边界的标签

有边界的标签



## 9.3 一般类控件（五）

### 标签（Label）（三）

- ✦ 用标签控件为没有Caption属性的控件创建快捷访问键
  - 将标签和要添加访问键的控件分别添加到窗体中
  - 将标签的TabIndex属性设置为控件的TabIndex属性减1
  - 在标签的Caption属性中将被作为访问键的字母前加上一个&符号。如果要在标签中显示&符号，需要将标签的UseMnemonic属性设置为False。运行时，带下划线的字母为访问键



## 9.3 一般类控件（六）

### 文本框（TextBox）（一）

✦ **基本功能：** 用来在运行时显示文本或接受程序使用人员输入的文本

#### ✦ 常用属性（一）

**Text属性：** 返回或设置显示或输入到文本框中的文本信息

（语法：`TextBox1.Text [=string]`）

**MultiLine属性：** 返回或设置文本框中是否可以输入多行文本

（语法：`TextBox1.MultiLine [=boolean]`）

**HideSelection属性：** 设置当文本框不具有焦点时，文本框中选择的文本是否仍然高亮度显示

（语法：`TextBox1.HideSelection [=boolean]`）



## 9.3 一般类控件（七）

### 文本框（TextBox）（二）

#### 常用属性（二）

**MaxLength属性：** 返回或设置文本框最多可容纳的字符数  
(语法: `TextBox1.MaxLength[=number]`)

**PasswordChar属性：** 设置文本框是否作为口令框  
(语法: `TextBox1.PasswordChar[=string]`)

**Locked属性：** 设置在运行时输入文本框的文本能否被编辑  
(语法: `TextBox1.Locked[=boolean]`)



## 9.3 一般类控件（八）

### 文本框（TextBox）（三）

#### 常用属性（三）

**ScrollBar属性：** 设置文本框是否具有滚动条  
(语法: `TextBox1.ScrollBar [=number]`)





## 9.3 一般类控件（九）

### 文本框（TextBox）（四）

#### ✦ 常用属性（四）

**SelStart属性：**返回或设置文本在文本框中的插入点  
（语法：`TextBox1.SelStart [=number]`）

**SelLength属性：**返回或设置文本框中缺省选中的字符数  
（语法：`TextBox1.SelLength [=number]`）

**SelText属性：**返回或设置文本框中当前被选中的文本  
（语法：`TextBox1.SelText [=string]`）

#### ✦ 常用事件过程

文本框控件支持Change、GotFocus和LostFocus等多个事件

**Change事件：**当文本框的内容发生改变时触发  
（语法：`Private Sub TextBox1_Change ()`）





## 9.3 一般类控件（十）

### 滚动条（ScrollBar）（一）

✦ **基本功能：** 用来滚动显示在屏幕上的内容，可分为水平滚动条（HScrollBar）和垂直滚动条（VScrollBar）

✦ **常用属性（四）**

**Value属性：** 返回或设置滚动框在滚动条中的位置

（语法： `ScrollBar1.Value[=number]`）

**LargeChange、SmallChange属性：** 设置用鼠标单击滚动箭头时滚动框每次移动的最大、最小距离

（语法： `ScrollBar1.LargeChange[=number]`

`ScrollBar1.SmallChange[=number]`）

**Max、Min属性：** 设置当滚动框位于水平滚动条最右端（最左端）或者垂直滚动条最下端（最上端）时的值

（语法： `ScrollBar1.Max[=number]`

`ScrollBar1.Min[=number]`）



## 9.3 一般类控件（十一）

### 滚动条（ScrollBar）（二）

#### 常用事件过程

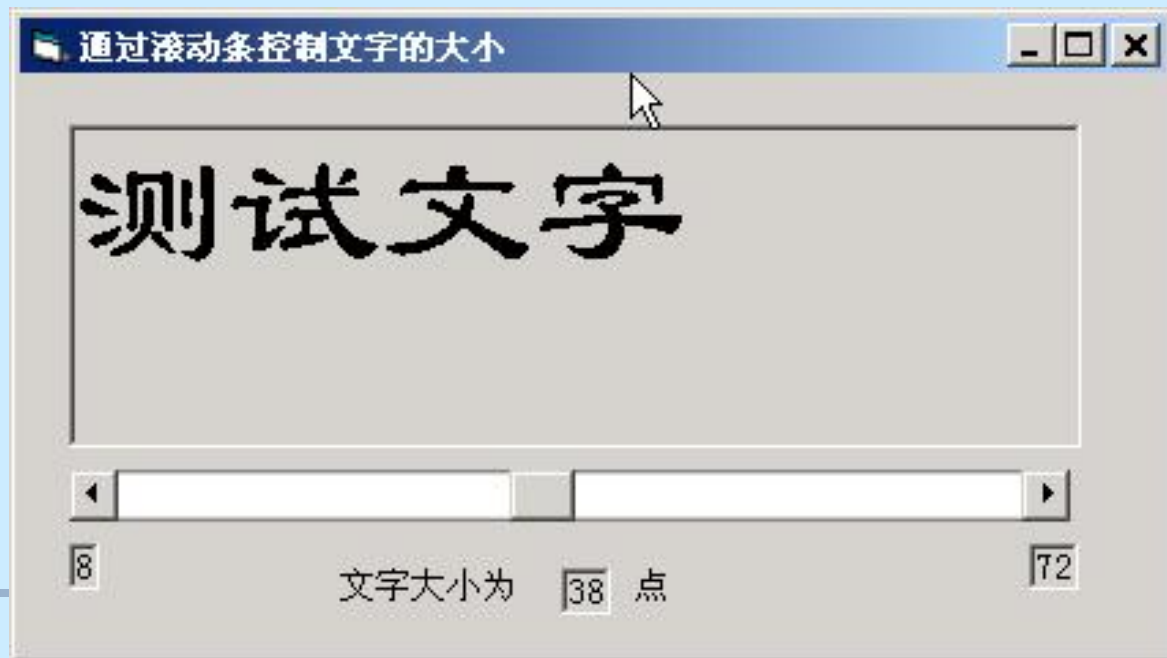
**Scroll事件：**当用鼠标拖动滚动框时触发

（语法：`Private Sub ScrollBar1_Scroll（）`）

**Change事件：**释放滚动框、单击滚动条或滚动箭头时触发

（语法：`Private Sub ScrollBar1_Change（）`）

**【例9.16】**通过使用水平滚动条控制标签内文字的大小



## 9.3 一般类控件（十二）

### 滚动条（ScrollBar）（三）

【例9.16】在本例中，控件定义如下：

用于显示文本的标签的Name属性为LblFont；

用于显示文本尺寸的标签的Name属性为LblSize；

滚动条的Name属性为HScr1Size，SmallChange属性值为1，LargeChange属性值为5，Min属性值为8，Max属性值为72。

```
Private Sub HScr1Size_Change ()  
    LblFont.FontSize = HScr1Size.Value  
    ' 根据滚动框在滚动条中的位置确定标签中文字的大小  
    LblSize.Caption = HScr1Size.Value  
    ' 根据滚动框在滚动条中的位置确定应显示的文字的尺寸  
End Sub
```



## 9.3 一般类控件（十三）

### 定时器（Timer）

✦ **基本功能：** 用来检查系统时钟，以确定是否执行某项操作

✦ **常用属性**

**Interval属性：** 返回或设置定时器事件之间的时间间隔  
(语法: `Timer1.Interval [=number]`)

**Enabled属性：** 用来决定该控件是否对时间做倒计时响应  
(语法: `ScrollBar1.LargeChange [=number]`  
`TextBox1.SetText [=string]`)

✦ **常用事件过程**

**Timer事件：** 当达到定时器控件的Interval属性规定的时间间隔时触发 (语法: `Private Sub Timer1_Timer()`)



## 9.3 一般类控件（十四）

### 数据（Data）控件

数据（Data）控件主要用来连接现有数据库，并将数据库中的信息显示在窗体中。

使用数据控件不用编写代码，就可以创建简单的数据库应用程序。

关于数据（Data）控件详细介绍，请参见数据库的相关讲解。

### OLE容器（OLE）控件

OLE容器（OLE）控件主要用来在Visual Basic应用程序中显示并操作其他基于Windows的应用程序（如Microsoft Excel和Microsoft Word for Windows）中的数据。



## 9.4 图形、图像类控件（一）

### 图片框（PictureBox）（一）

✦ **基本功能：** 用来显示图片或作为其他控件的容器

✦ **常用属性**

**Picture属性：** 保存和设置显示在图片框中的图片  
(语法: `PictureBox1.Picture[=picture]`)

**Align属性：** 设置图片框在窗体中的显示方式  
(语法: `PictureBox1.Align[=number]`)

**AutoSize属性：** 决定图片框控件是否能够根据装载的图片的大小自动调整尺寸 (语法: `PictureBox1.AutoSize[=boolean]`)

**Picture属性：** 保存和设置显示在图片框中的图片  
(语法: `PictureBox1.Picture[=picture]`)



## 9.4 图形、图像类控件（二）

### 图片框（PictureBox）（二）

✦ **常用事件过程：**可以响应Click事件

✦ **常用方法**

**Print方法：**用于在图片框中显示文本  
(语法: PictureBox1.Print[string])

**Circle方法：**用于在图片框中绘制圆形  
(语法: PictureBox1.Circle (X, Y) , r)

**【例9.18】**下面语句用来在图片框Mypicture中绘制一个圆，该圆的圆心坐标为（500，500），半径为1000点。

```
Mypicture.AutoRedraw=True
```

```
Mypicture.Circle (500, 500) , 1000
```





## 9.4 图形、图像类控件（三）

### 图像（Image）

- ✦ **基本功能：** 用来显示图像

- ✦ **常用属性**

**Picture属性：** 保存和设置显示在图像框中的图形  
(语法: `Image1.Picture[=picture]`)

**Stretch属性：** 决定图像控件与被装载的图片如何调整尺寸以互相适应  
(语法: `Image1.Stretch[=boolean]`)

- ✦ **常用事件过程：** 可以响应Click事件



## 9.4 图形、图像类控件（四）

### 形状（Shape）

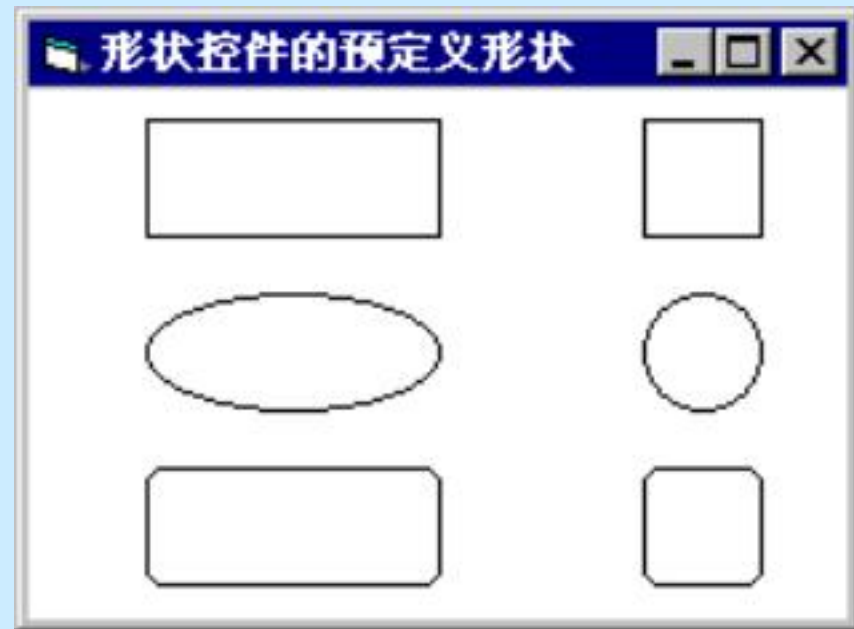
✦ **基本功能：** 用于在窗体、框架或图片框中绘制预定义的几何图形

✦ **常用属性**

**Shape属性：** 返回或设置形状控件的外观

（语法：`Shape1.Shape [=number]`）

| 取 值                       | 含 义   |
|---------------------------|-------|
| 0-vbShapeRectangle        | 矩形    |
| 1-vbShapeSquare           | 正方形   |
| 2-vbShapeOval             | 椭圆形   |
| 3-vbShapeCircle           | 圆形    |
| 4-VbShapeRoundedRectangle | 圆角矩形  |
| 5-VbShapeRoundedSquare    | 圆角正方形 |



## 9.4 图形、图像类控件（五）

### 线形 (Line)

✦ **基本功能：** 用来在窗体、框架或者图片框的表面绘制简单的线段

✦ **常用属性**

**X1, Y1, X2, Y2属性：** 通过设置线段的起点坐标 (X1, Y1) 和终点坐标 (X2, Y2) 属性，可以设置线段的长度

(语法: `Line1.X1 [=number]`  
`Line1.Y1 [=number]`  
`Line1.X2 [=number]`  
`Line1.Y2 [=number]`)



## 9.5 选择类控件（一）

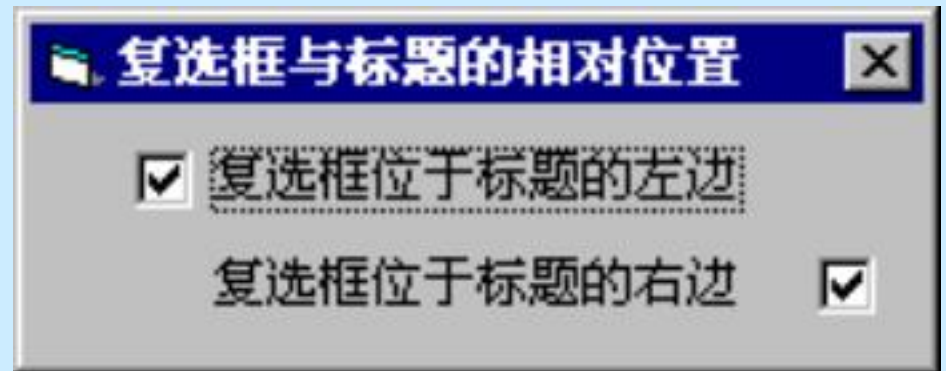
### 复选框（CheckBox）（一）

✚ **基本功能：** 用来表示Yes/No或True/False等状态

✚ **常用属性**

**Value属性：** 返回或设置复选框所处的状态  
(语法: `CheckBox1.Value[=number]`)

**Alignment属性：** 返回或设置复选框与Caption属性所设置的标题的相对位置 (语法: `CheckBox1.Alignment[=number]`)



## 9.5 选择类控件（二）

### 复选框（CheckBox）（二）

#### ✦ 常用事件过程

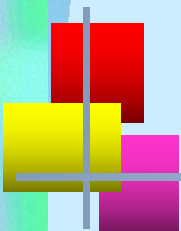
**Click事件：**有如下几种触发条件

用鼠标单击复选框控件

在键盘上按Tab键，将焦点转移到复选框控件上时按Spacebar（空格）键

在Caption属性的一个字母前加连字符（&），创建一个访问键，通过Alt+“访问键”切换复选框控件的选择

在程序代码中将复选框的Value属性设置为True



## 9.5 选择类控件（三）

### 框架（Frame）

✦ **基本功能：** 用来为其他控件提供可标识的分组

#### ✦ 常用属性

**Caption属性：** 返回或设置框架的标题名称  
(语法: `Frame1.Caption[=string]`)

**Enabled属性：** 返回或设置框架的活动状态  
(语法: `Frame1.Enabled[=boolean]`)



## 9.5 选择类控件（四）

### 选项按钮（OptionButton）（一）

✦ **基本功能：** 用来显示选项，通常以按钮组的形式出现

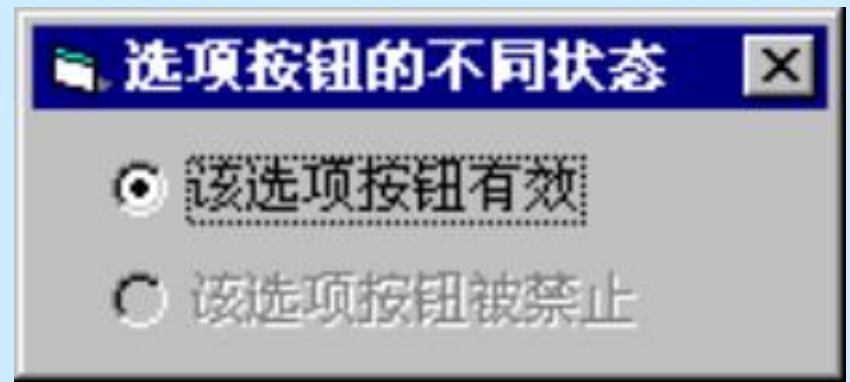
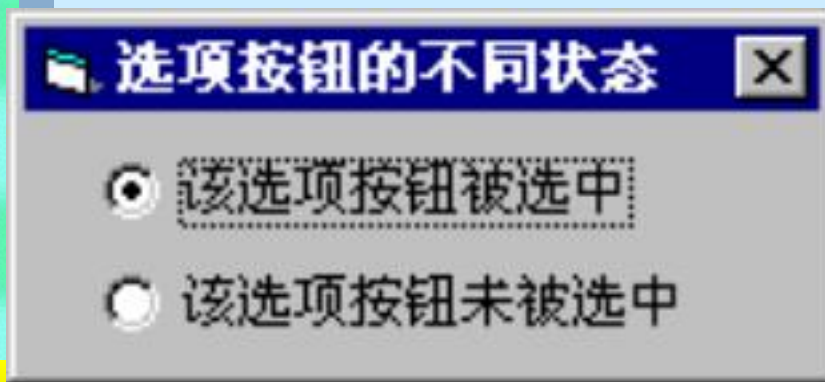
✦ **常用属性**

**Value属性：** 返回或设置选项按钮的状态

（语法：`OptionButton1.Value[=boolean]`）

**Enabled属性：** 返回或设置选项按钮是否被禁止

（语法：`OptionButton1.Enabled[=boolean]`）





## 9.5 选择类控件（五）

### 选项按钮（OptionButton）（二）

#### ✦ 创建选项按钮组的步骤

- 绘制框架或图片框
- 将同一组的选项按钮放到同一个框架或图片框中
- 在设计时，框架或图片框与其中的选项按钮将作为一个整体移动

**【例9.26】** 图中有两组选项按钮，其中一组直接添加到窗体中，另一组添加到框架中。



直接添加到窗体中的选项按钮组

框架中的选项按钮组



## 9.5 选择类控件（六）

### 列表框（ListBox）（一）

✦ **基本功能：** 用来以项目列表形式显示一系列项目，并可以从中选择一项或多项

#### ✦ 常用属性（一）

**Columns属性：** 返回或设置列表项显示的列数  
(语法: `ListBox1.Columns [=number]`)

**List属性：** 设置或返回列表框控件的指定列表项  
(语法: `ListBox1.List (index) [=string]`)

**ItemData属性：** 返回或设置列表框控件中每个列表项具体的编号  
(语法: `ListBox1.ItemData (index) [=number]`)



## 9.5 选择类控件（七）

### 列表框（ListBox）（二）

#### ✦ 常用属性（二）

**ListCount属性：**返回列表框中列表项的数目  
（语法：`ListBox1.ListCount`）

**ListIndex属性：**返回或设置列表框控件中当前选择的列表项的索引  
（语法：`ListBox1.ListIndex[=number]`）

**NewIndex属性：**返回最后添加到列表框中的列表项的索引  
（语法：`ListBox1.NewIndex`）

**Text属性：**返回或设置列表框控件中当前选择的列表项的内容  
（语法：`ListBox1.Text[=string]`）

**Sorted属性：**返回或设置列表框控件的元素是否自动按字母表顺序排序  
（语法：`ListBox1.Sorted[=boolean]`）



## 9.5 选择类控件（八）

### 列表框（ListBox）（三）

#### 常用属性（三）

**MultiSelect属性：**返回或设置是否能够在列表框控件中进行复选，以及如何进行复选（语法：`ListBox1.MultiSelect[=number]`）

MultiSelect属性取  
值为0的列表框



MultiSelect属性取  
值为1的列表框

MultiSelect属性取  
值为2的列表框



## 9.5 选择类控件（九）

### 列表框（ListBox）（四）

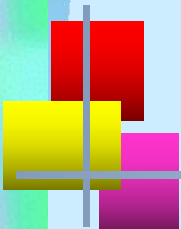
✦ **常用事件过程：** 支持Click、Db1Click、MouseDown、MouseUp、MouseMove、GotFocus、LostFocus等事件

#### ✦ 常用方法

**AddItem方法：** 用于在列表框控件中添加列表项  
(语法: `ListBox1.AddItem string[, number]`)

**RemoveItem方法：** 用于从列表框控件中删除指定列表项  
(语法: `ListBox1.RemoveItem number`)

**Clear方法：** 用于删除列表框中的所有列表项  
(语法: `ListBox1.Clear`)



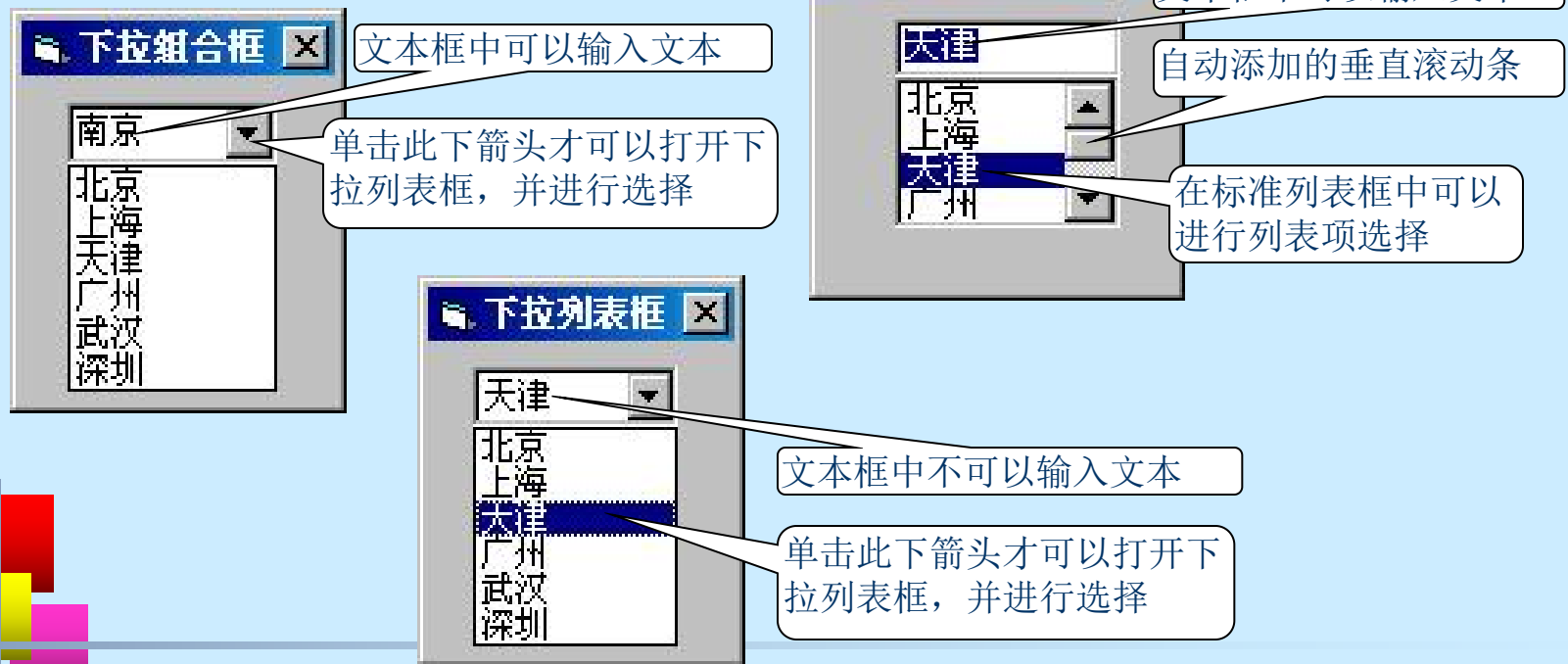
## 9.5 选择类控件（十）

### 组合框（ComboBox）（一）

✦ **基本功能：** 将文本框控件和列表框控件的特性结合在一起，可以在控件的文本框中输入文本或从列表框中选择列表项

#### ✦ 常用属性（一）

**Style属性：** 返回或设置组合框控件的显示类型和方式  
(语法: `ComboBox1.Style`)





## 9.5 选择类控件（十一）

### 组合框（ComboBox）（二）

#### ✦ 常用属性（二）

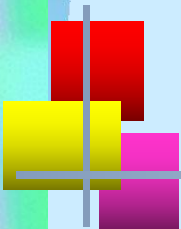
**Text属性：**用来返回或设置文本框中的文本或选择的列表项的内容  
(语法: `ComboBox1.Text [=string]`)

✦ **常用事件过程：**支持Click、DoubleClick、MouseDown、MouseUp、MouseMove、GotFocus、LostFocus等事件

#### ✦ 常用方法

**AddItem方法：**用于在列表框控件中添加列表项

**RemoveItem方法：**用于从列表框控件中删除指定列表项





## 9.6 控件数组

控件数组是具有共同名称、类型和事件过程的一个或多个控件。

### 控件数组的作用

- ✦ **在设计时**，通过控件数组添加控件所消耗的资源比直接向窗体添加多个相同类型的控件消耗的资源要少得多。另外，如果有多个控件共享某一相同代码时，使用控件数组也很方便
- ✦ **在运行时**，如果想创建新的控件，则必须使用控件数组。新控件必须是某一控件数组的成员，每个新控件都将继承为控件数组编写好的公共事件过程

### 控件数组的创建

#### ✦ 在设计时创建的方法

- 将同一名字赋给多个相同类型的控件
- 复制现有控件
- 将控件的Index属性设为非Null的数值

- ✦ **在运行时**，可以通过Load和Unload语句添加和删除控件数组中的控件



# 思考与练习

## 填空题

- ✦ 对于某一对象能否接收焦点，取决于该对象的\_\_\_\_\_和属性的取值。\_\_\_\_\_属性允许对象响应键盘、鼠标等事件。\_\_\_\_\_属性则决定对象是否显示在屏幕上。只有这两个属性的取值同时均为\_\_\_\_\_时，该对象才能接收焦点。
- ✦ Tab键的顺序就是当按下\_\_\_\_\_键时，焦点在窗体中的各控件间移动的顺序。每个窗体都具有相应的Tab键的顺序。在缺省情况下，Tab键的顺序与\_\_\_\_\_顺序相同。



# 思考与练习

## 简答题

- ✦ Visual Basic的控件可分为哪3种类型？简述各种类型控件的特点。
- ✦ 在设计时如何使用控件？
- ✦ 为控件命名应遵循哪些原则？
- ✦ 将焦点赋给某一对象有哪些方法？有哪些控件不能接收焦点？
- ✦ 如何创建控件数组？
- ✦ 简述各内部控件的功能。
- ✦ 试运行本章中各个例题。

