

第8章 窗体

知识要点

- Visual Basic窗体的常用属性、事件和方法
- 使用窗体进行Visual Basic应用程序界面设计的方法
- 使用多个窗体进行Visual Basic应用程序界面设计的方法
- 多文档界面（MDI）窗体的相关知识

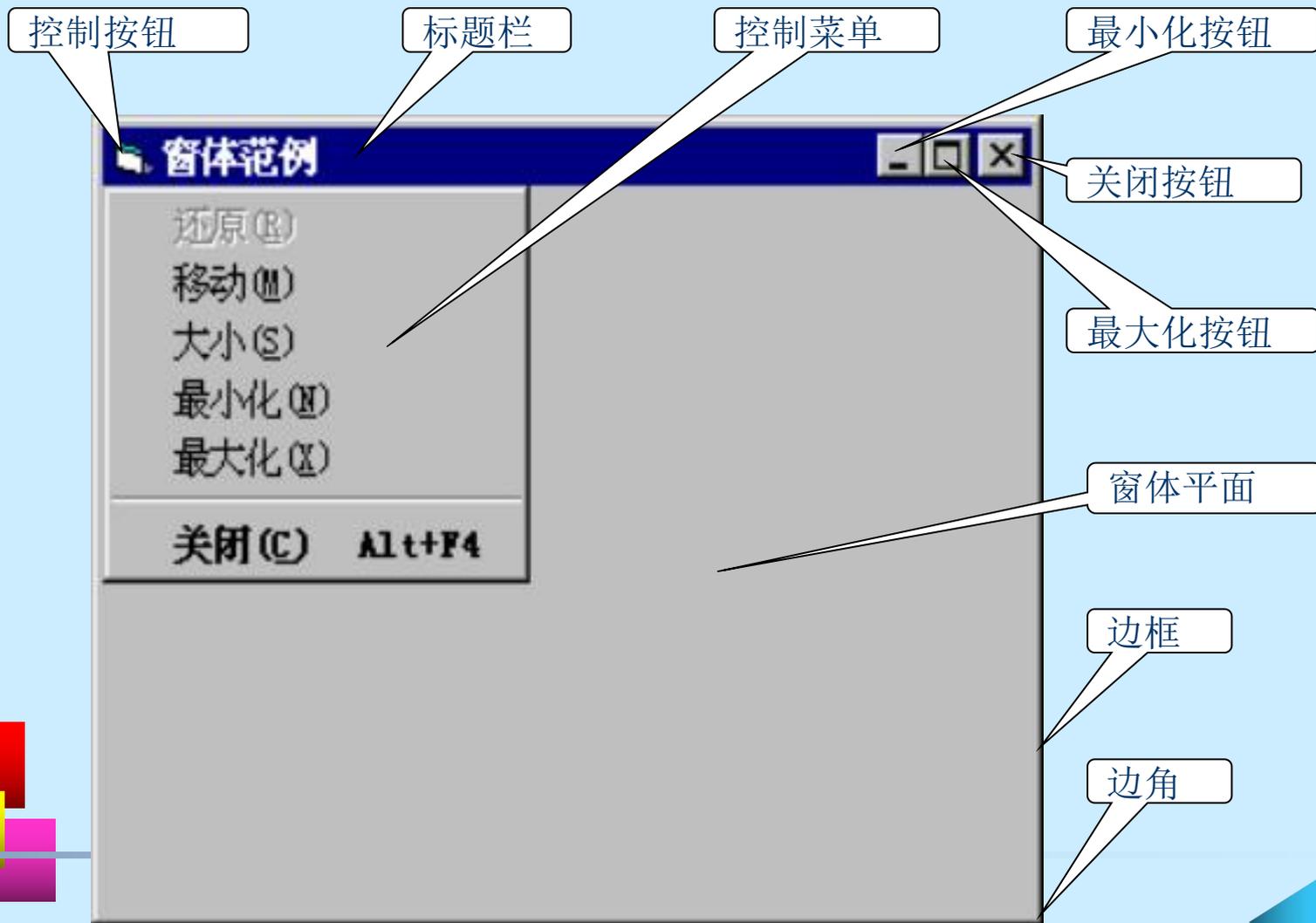
学习任务

- 了解多文档界面（MDI）窗体的相关知识
- 理解如何在一个Visual Basic应用程序中使用多个窗体进行界面设计
- 掌握Visual Basic窗体的常用属性、事件和方法，以及使用窗体进行Visual Basic应用程序界面设计的方法



8.1 窗体简介

窗体是Visual Basic应用程序运行界面的重要组成部分，任何一个应用程序都至少有一个窗体，窗体本身也是一种对象。



8.2 窗体的属性（一）

窗体属性的设置方法

- ✦ 在窗体上单击鼠标右键，在弹出的菜单中单击“属性窗口”选项或单击窗体并按下F4键来激活“属性”窗口，在“属性”窗口中可以设置窗体的属性
- ✦ 在程序运行时由程序代码来实现窗体的属性设置

窗体的常用属性(一)

- ✦ **Name属性**：返回或设置在程序代码中用于标识窗体的名字，该属性在运行时不可见（语法：`object.Name`）
- ✦ **Appearance属性**：返回或设置窗体或窗体上的控件的显示效果（语法：`object.Appearance [=Value]`）

Appearance属性的设置值是：

- 0 窗体及窗体上的控件显示为平面效果
- 1 窗体及窗体上的控件显示为立体效果，为缺省值。



8.2 窗体的属性（二）

窗体的常用属性（二）

- ✦ **AutoRedraw属性**：返回或设置对象的自动重绘是否有效（语法：`object.AutoRedraw[=boolean]`）

AutoRedraw属性的设置值是：

True 使对象的自动重绘有效。此时对象显示到屏幕上

False 缺省值，使对象的自动重绘无效

- ✦ **BackColor属性和ForeColor属性**：返回或设置对象的背景颜色或对象中显示的图片 and 文本的前景颜色

(语法：`object.BackColor[=color]`或`object.ForeColor[=color]`)

常数	值	对应颜色
vbBlack	0x0	黑色
vbRed	0xFF	红色
vbGreen	0xFF00	绿色
vbYellow	0xFFFF	黄色
vbBlue	0xFF0000	蓝色
vbMagenta	0xFF00FF	紫红色
vbCyan	0xFFFF00	青色
vbWhite	0xFFFFFFFF	白色

8.2 窗体的属性（三）

窗体的常用属性（三）

✦ **BorderStyle属性：** 返回或设置对象的边框样式
(语法: `object.BorderStyle[=value]`)

BorderStyle属性的设置值是：

- 0 无。没有边框及与边框相关的元素
- 1 固定单边框
- 2 可调整的边框（缺省值）
- 3 固定对话框
- 4 固定工具窗口
- 5 可变尺寸工具窗口

✦ **Caption属性：** 设置显示在窗体的标题栏中的文本
(语法: `object.Caption[=string]`)



8.2 窗体的属性（四）

窗体的常用属性（四）

- ✦ **ClipControls属性：** 返回或设置一个值，指定Paint事件中的图形方法的影响范围

（语法：`object.ClipControls[=boolean]`）

ClipControls属性的设置值是：

`True` 指定Paint事件中的图形方法绘制整个窗体

`False` 指定Paint事件中的图形方法只绘制窗体中新显露出的区域

- ✦ **ControlBox属性：** 返回或设置一个值，指示在程序运行时窗体中是否显示控制菜单框

（语法：`object.ControlBox[=boolean]`）

ControlBox属性的设置值是：

`True` 显示控制菜单框（缺省值）

`False` 不显示控制菜单框



8.2 窗体的属性（五）

窗体的常用属性（五）

- ✦ **Enabled属性：** 返回或设置窗体是否能够对键盘或鼠标产生的事件做出反应（语法：`object.Enabled[=boolean]`）

Enabled属性的设置值是：

True 设置object对事件做出反应（缺省值）

False 设置object对事件不做反应

- ✦ **Font属性：**

返回一个决定在窗体中显示的文本所使用的字体Font对象

（语法：`object.Font`）



8.2 窗体的属性（六）

窗体的常用属性（六）

- ✦ 在**Height、Width属性**：返回或设置窗体的高度和宽度
(语法：`object.Height[=number]`和`object.Width[=number]`)
 - ✦ **Icon属性**：返回或设置程序运行时窗体处于最小化状态或在窗体的左上角看到窗体的图标。
(语法：`object.Icon[=LoadPicture(iconpath)]`)
 - ✦ **Left和Top属性**：返回或设置窗体内部的左边或内侧顶边与它的容器的左边或顶边之间的距离
(语法：`object.Left[=value]`和`object.Top[=value]`)
- 【例 8.1】**编写程序代码：在窗体被加载时将窗体面积设置为屏幕大小的一半，并使窗体位于屏幕中央。

```
Private Sub Form_Load()  
    Width=Screen.Width * .5           ' 设置窗体的宽度  
    Height=Screen.Height * .5        ' 设置窗体的高度  
    Left=(Screen.Width-Width)/2      ' 设置窗体在水平方向上居中  
    Top=(Screen.Height-Height)/2    ' 设置窗体在垂直方向上居中  
End Sub
```



8.2 窗体的属性（七）

窗体的常用属性（七）

- ✦ **Moveable属性：**返回或设置窗体的位置是否可以被移动
(语法: `object.Moveable=boolean`)

Moveable属性的设置值是:

True或-1 窗体的位置可以被移动

False或0 窗体的位置不能被移动

- ✦ **MaxButton、MinButton属性：**返回一个值，确定窗体的“最大化”或“最小化”按钮是否有效

(语法: `object.MaxButton`和`object.MinButton`)

MaxButton或MinButton属性的设置值是:

True 窗体的“最大化”或“最小化”按钮有效（缺省值）

False 窗体的“最大化”或“最小化”按钮无效



8.2 窗体的属性（八）

窗体的常用属性（八）

✦ **Picture属性：** 返回或设置窗体中显示的图片

（语法：`object.Picture[=picture]`）

`Picture`用于指定图片文件，设置如下：

`None` 代表没有图片（缺省值）

`Bitmap`、`icon`、`metafile`、`GIF`、`JPEG` 可以指定一个图片

【例】`Form1.Picture = LoadPicture("c:\windows\clouds.bmp")`

✦ **StartPosition属性：** 返回或设置窗体首次出现时的显示位置
（语法：`object. [=position]`）

`StartPosition`属性的设置值是：

0 手动指定取值，窗体的初次显示位置由`Left`和`Top`属性决定

1 所隶属的对象的中央

2 屏幕中央

3 窗口缺省（屏幕的左上角）



8.2 窗体的属性（九）

窗体的常用属性（九）

ScaleLeft和ScaleTop属性：返回或设置窗体的左边界的水平坐标或上边界的垂直坐标

（语法：`object.ScaleLeft[=value]`和`object.ScaleTop[=value]`）

✚ **Visible属性：**返回或设置窗体首次出现时的显示位置

（语法：`object.Visible[=boolean]`）

Visible属性的设置值是：

True 设置窗体是可见的（缺省值）

False 设置窗体是隐藏的

【例8.2】当单击窗体时，相应窗体通过设置Visible属性被隐藏，并在屏幕上显示提示信息。按照提示信息单击“确定”按钮，该窗体通过设置Visible属性重新显示在屏幕上。

```
Private Sub Form_Click()  
    Visible=false           '通过设置Visible属性隐藏窗体  
    MsgBox “单击“确定”可以显示窗体。”    '显示提示信息  
    Visible=true           '通过设置Visible属性显示窗体  
End Sub
```



8.2 窗体的属性（十）

窗体的常用属性（十）

- ✦ **WindowState属性**：返回或设置一个数值，用来指定窗体的可视状态

（语法：`object.WindowState[=value]`）

WindowState属性的设置值是：

- 0 窗体以正常方式显示，为缺省取值
- 1 窗体缩到最小，显示为图标状态
- 2 窗体放大到最大尺寸



8.2 窗体的属性（十一）

通过属性设计窗体的外观

✦ 与窗体外观有关的属性

Appearance（立体显示）

BackColor（背景颜色）

ForeColor（前景颜色）

BorderStyle（边框样式）

Caption（标题）、Font（显示字体）

Height（高度）、Width（宽度）

Left（左边距）、Top（上边距）

ControlBox（控制按钮）

MaxButton、MinButton（“最大化”、“最小化”按钮）

WindowState（可视状态）

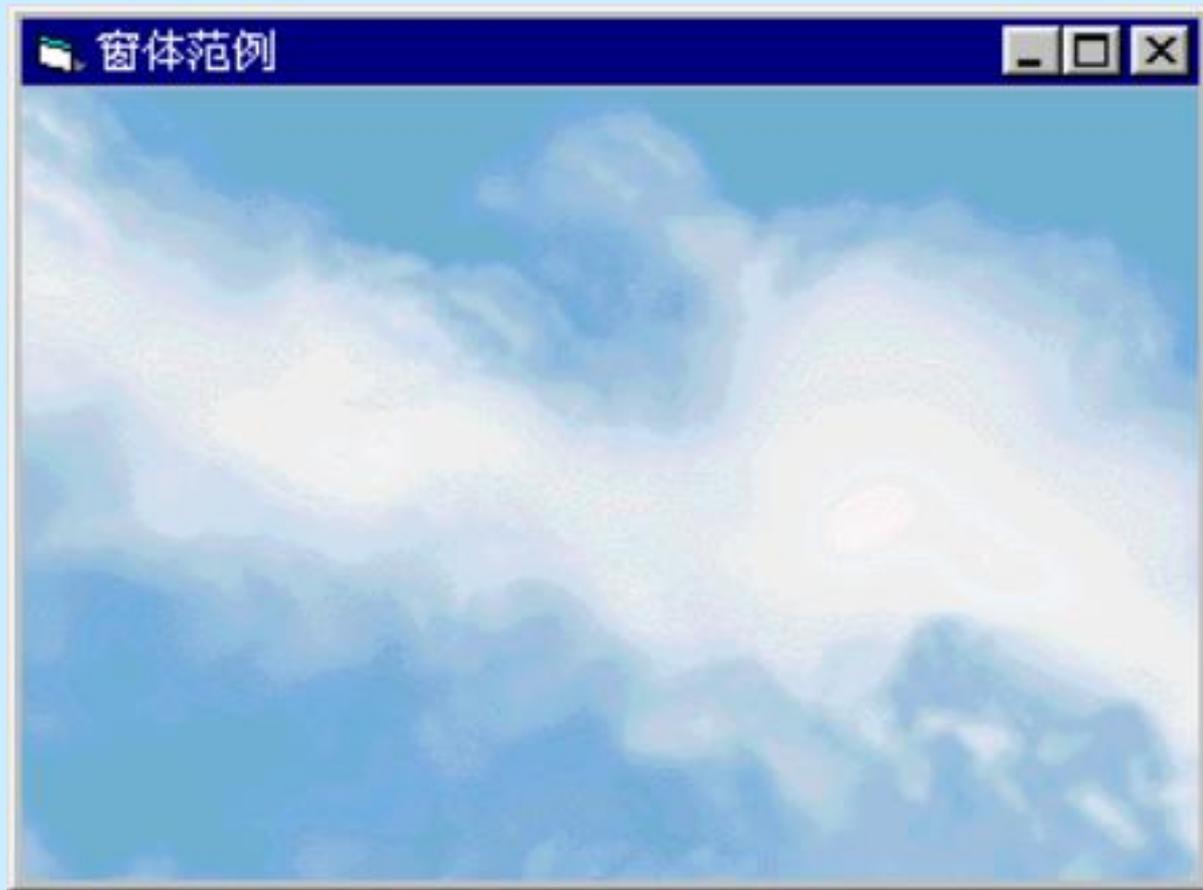
StartPosition（起始位置）



8.2 窗体的属性（十二）

✦ 窗体外观设计实例

【例8.3】建立一个窗体，并对其进行属性设置，使该窗体的外观符合Windows应用程序界面风格。



8.2 窗体的属性（十三）

✦ 窗体外观设计实例——操作步骤

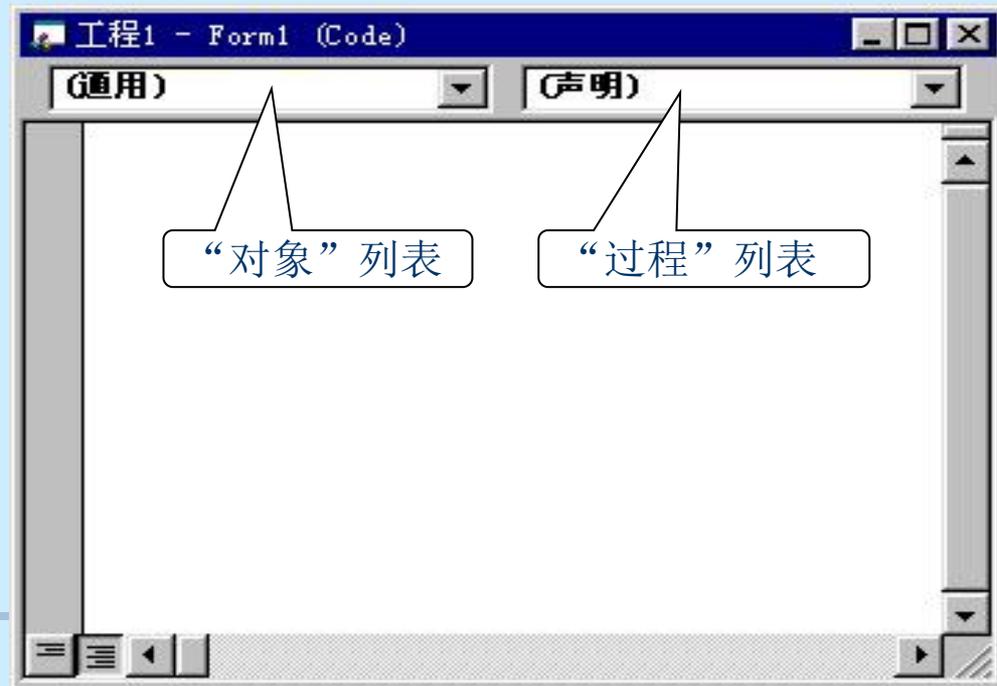
- 创建窗体
- 设置窗体的Caption（标题）属性为“窗体范例”
- 设置窗体首次出现时的显示位置：设置Left（左边距）属性和Top（上边距）属性分别为4000和2500
- 设置窗体首次出现时的大小：设置Height（高度）和Width（宽度）属性分别为7000和9300
- 设置窗体的Appearance（立体显示）属性为1。
- 设置窗体的BorderStyle（边框样式）属性的取值为2（即可调整的边框）
- 设置窗体显示控制菜单框、“最大化”按钮和“最小化”按钮为可见：设置ControlBox（控制按钮）属性的取值为True
- 设置窗体的MaxButton（“最大化”按钮）属性和MinButton（“最小化”按钮）属性均为True
- 设置窗体的背景图案Picture属性为“C:\WINDOWS\CLOUDS.BMP”
- 设置窗体的字体样式Font属性为“宋体”，字号为“小五”



8.3 窗体的事件（一）

编写窗体的事件过程代码

- 单击要编写事件过程的窗体
- 在“视图”菜单中选择“代码窗口”选项，或在“工程资源管理器”窗口中单击左上角的“查看代码”按钮，调出代码窗口
- 单击代码窗口标题条下面的“对象”下拉列表并选择相应窗体
- 单击“过程”下拉列表并选择相应的事件名称
- 编写窗体的事件过程代码



8.3 窗体的事件（二）

窗体的常用事件过程（一）

✚ **Click事件**：当用鼠标单击窗体的空白区域或窗体上的一个无效控件被触发（语法：`Private Sub Form_Click()`）

【例8.4】编写一段程序代码实现以下功能：每次用鼠标单击窗体，该窗体的面积将会变大。

```
Private Sub Form_Click()  
    Print "窗体发生鼠标单击（Click）事件，窗体将变大"  
    Form1.Height = Form1.Height + 60  
    Form1.Width = Form1.Width + 80  
End Sub
```

✚ **DbClick事件**：当用鼠标双击窗体的空白区域或窗体上的一个无效控件时被触发（语法：`Private Sub Form_DbClick()`）

✚ **Initialize事件**：当应用程序创建一个窗体时被触发
（语法：`Private Sub Form_Initialize()`）

✚ **Load事件**：当窗体被装载时被触发
（语法：`Private Sub Form_Load()`）



8.3 窗体的事件（三）

窗体的常用事件过程（二）

- ✦ **QueryUnload事件**：当窗体将要关闭时被触发
(语法：Private Sub Form_QueryUnload(cancel As Integer,unloadmode As Integer))
- **Unload事件**：当窗体从屏幕上删除时被触发
(语法：Private Sub Form_Unload(cancel As Integer))
- **MouseMove事件**：当鼠标移动时被触发（语法：Private Sub Form_MouseMove (button As Integer,shift As Integer,x As Single,y As Single)）
- ✦ **Activate和Deactivate事件**：当窗体成为活动窗口或变为非活动窗口时被触发
(语法：Private Sub Form_Activate()和Private Sub Form_Deactivate())
- ✦ **GotFocus事件**：当窗体获得焦点时被触发
(语法：Private Sub Form_GotFocus())
- **LostFocus事件**：当窗体失去焦点时被触发
(语法：Private Sub Form_LostFocus())



8.3 窗体的事件（四）

窗体的常用事件过程（三）

- ✦ **MouseDown和MouseUp事件：**当按下或松开鼠标按钮时被触发
(语法: `Private Sub Form_MouseDown(button As Integer,shift As Integer,x As Single,y As Single)`)

【例8.5】编写一段程序代码实现以下功能：每次用鼠标单击窗体，如果按下的是左键则该窗体的位置将会向左移动，如果按下的是右键则该窗体的位置将会向右移动。

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer,
    X As Single, Y As Single)
    If Button = 1 Then                                '如果在窗体上按下了鼠标左键
        Print " 在窗体上按下了鼠标左键，窗体将向左移动" '显示提示信息
        Form1.Left = Form1.Left - 100                '窗体将向左移动
    End If
    If Button = 2 Then                                '如果在窗体上按下了鼠标右键
        Print "在窗体上按下了鼠标右键，窗体将向右移动"
        Form1.Left = Form1.Left + 100
    End If
End Sub
```



8.4 窗体的方法

窗体的常用方法

- ✦ **Load语句**: 将窗体加载到内存中 (语法: `Load object`)
- ✦ **Show方法**: 用于窗体的显示 (语法: `object.Show`)
- ✦ **Hide方法**: 用于隐藏窗体 (语法: `object.Hide`)
- ✦ **Unload语句**: 用于从内存中卸载窗体 (语法: `Unload object`)
- ✦ **Move方法**: 用于移动窗体
(语法: `object.Move left, top, width, height`)
- ✦ **Refresh方法**: 用于对一个窗体进行全部重绘
(语法: `object.Refresh`)
- ✦ **SetFocus方法**: 用于使窗体获得焦点 (语法: `object.SetFocus`)
- ✦ **PopupMenu方法**: 在窗体上的指定位置或鼠标当前位置显示弹出式菜单 (语法: `[object.]PopupMenu menuname [, flags[, x[, y [, boldcommand]]]]`)



8.5 多个窗体的处理（一）

加入新的窗体

- 在“文件”菜单中选取“新建工程”创建一个工程，或选取“打开工程”打开一个已有的工程
- 在“工程”菜单中单击“添加窗体”选项
- 在“新建”中选择要添加的窗体的类型，或者在“现存”中选择一个已经存在的窗体文件
- 单击“打开”按钮



8.5 多个窗体的处理（二）

设定启动窗体

- ✦ 将其他窗体作为应用程序的启动窗体的操作步骤：
 - 在“工程”菜单中，选取最后一个选项“工程属性”
 - 在弹出的“工程属性”对话框中单击“通用”
 - 在“启动对象”下拉列表框中，选取一个窗体作为启动窗体
 - 单击“确定”按钮
- ✦ 动态设置启动窗体：在标准模块中建立一个名为Main的子过程，可以动态地显示窗体，步骤如下：
 - 在“工程”菜单中，选取最后一个选项“工程属性”
 - 在弹出的“工程属性”对话框中单击“通用”
 - 在“启动对象”下拉列表框中，选取“Sub Main”
 - 单击“确定”按钮。
- ✦ 快速显示窗体：用于显示应用程序的名称、版权信息等内容

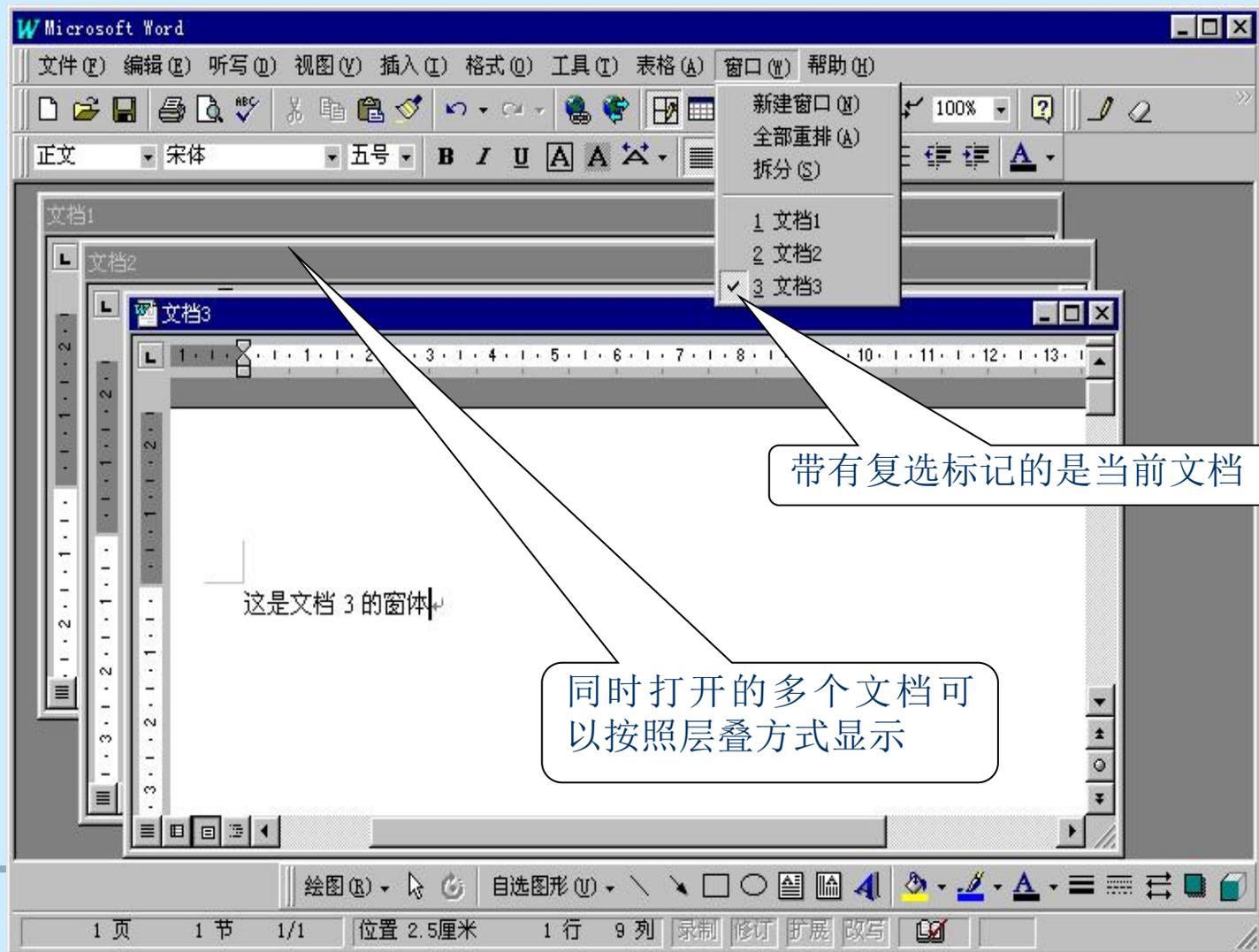
结束应用程序时关闭所有窗体

要结束一个应用程序，必须保证该应用程序中的所有窗体（都已经被关闭



8.6 多文档界面（MDI）窗体（一）

Windows应用程序主要有单文档界面（SDI）和多文档界面（MDI）两种界面样式。



8.6 多文档界面 (MDI) 窗体 (二)

建立多文档界面 (MDI) 应用程序 (一)

- ✦ **创建MDI窗体。** MDI窗体通常可以作为应用程序背景窗口
 - 在“文件”菜单中选取“新建工程”创建一个工程，或选取“打开工程”打开一个已有的工程
 - 在“工程”菜单中单击“添加MDI窗体”选项
- ✦ **建立MDI子窗体**
 - 建立一个新窗体或者打开一个已经存在的窗体
 - 将该窗体的MDIChild属性设置为True
- ✦ **多文档界面 (MDI) 窗体的加载与显示**
 - ✦ 在代码中引用一个窗体的属性将使该窗体被自动装载
 - ✦ 将MDI窗体的AutoShowChildren属性设置为True，则在该MDI窗体被加载时，其子窗体将自动显示；将MDI窗体的AutoShowChildren属性设置为False，则在该MDI窗体被加载时，其子窗体将自动隐藏
- ✦ 所有MDI子窗体被加载时都是可视的



8.6 多文档界面（MDI）窗体（三）

建立多文档界面（MDI）应用程序（二）

✦ 多文档界面（MDI）窗体的大小和位置

- ✦ 当MDI子窗体的边框大小可变时，其初始大小及显示位置由MDI窗体的大小决定。当MDI子窗体的边框大小不可变时，该子窗体的初始大小由设计时的Height属性和Width属性的取值决定

- ✦ MDI子窗体的设计与MDI窗体无关

- ✦ MDI子窗体被最小化时会以图标的形式显示在MDI窗体中，但并不显示在Windows的任务栏中

✦ 多文档界面（MDI）窗体的控件

只有具有Align（显示方位）属性的控件或者具有不可见界面的控件，才能直接放置在MDI窗体中



思考与练习

填空题

- ✦ 窗体通常有以下几个基本组成部分_____、_____、_____、_____、_____、_____和_____。
- ✦ 窗体本身是一种对象，可以通过_____定义窗体的外观，通过_____定义窗体的行为，通过_____定义窗体与程序使用者之间的交互。
- ✦ 每当建立一个工程文件时Visual Basic都会给出一个缺省名为_____的窗体。
- ✦ 名称属性必须以一个_____开始并且最长可达_____个字符，不能包括_____和_____。
- ✦ Caption属性可以设置显示在窗体的_____中的文本。



思考与练习

简答题

- ✦ 简述窗体属性的设置方法。
- ✦ 简述窗体的常用属性。
- ✦ 如何通过属性设计窗体的外观，请举例说明。
- ✦ 怎样进入窗体事件过程代码的编写状态？
- ✦ 简述窗体的几种常见事件。
- ✦ 如何进行窗体的加载、显示、隐藏和卸载？
- ✦ 如何为应用程序加入新的窗体？
- ✦ 如何设定启动窗体？
- ✦ 举例说明什么是多文档界面（MDI）。

