

学生应知信息知识

Visual Basic 简明教程 (中)

秋登峰 主编

# 目 录

第四章	Windows 用户窗体的设计 .....	1
4.1	VCL 控件窗体设计 .....	1
4.2	多重窗体设计 .....	9
4.2.1	多重窗体设计常用方法 .....	9
4.2.2	多重窗体设计实例 .....	10
4.3	多文档界面窗体设计 .....	28
4.4	小 结 .....	45
第五章	格式文件操作和处理 .....	45
5.1	文件的格式处理 .....	46
5.2	创建查询文件程序 .....	62
5.3	利用文件函数 .....	76
5.4	小 结 .....	92
第六章	多媒体程序开发和应用 .....	92
6.1	多媒体动画 .....	92
6.2	制作视频播放器 .....	110
6.3	制作音频播放器 .....	133
6.4	MCIWnd 多媒体制作 .....	147
6.5	小 结 .....	162

## 第四章 Windows 用户窗体的设计

用户窗体的界面设计可以分为三个方面的内容：第一个是利用 VCL 控件（可视化控件）的窗体界面设计，另外一个多重窗体的设计（Multiple Forms），还有一个就是经常提到的多文档界面的设计，即 MDI。

### 注意：

我们重点需要了解多重窗体和多文档界面窗体的区别。

多重窗体（也叫复合窗体）是指在一个应用程序中有多个窗口界面，一个个分别的显示在屏幕上，而多文档界面是指在一个父窗口中包含有多个子窗口，在程序运行过程中可以同时打开多个窗体，这些窗体之间的关系由程序指定。

下面将分别对三种窗体的界面设计方法加以介绍。

### 4.1 VCL 控件窗体设计

一个应用程序的好与坏，除了算法的设计，代码的编写方面的因素外，用户界面的设计也是一个非常重要的方面，在可视化的编程环境中，虽然不用编写大量的代码就可以设计出丰富的界面，但是要想设计出有个性有特色的用户界面，还是需要程序的开发人员下一番功夫的。

下面就是一个利用 VCL 控件来进行程序设计的示例，在这个示例程序中所用到的一些控件在程序的初始阶段没有被加载，还需要读者自己把它们添加到工具箱

中。

示例程序的具体步骤如下：

#### 1. 启动一个新项目

选择菜单“工程”/“部件”，在弹出的对话框中选择所需要的，添加控件后的工具箱如图 4-1 所示。

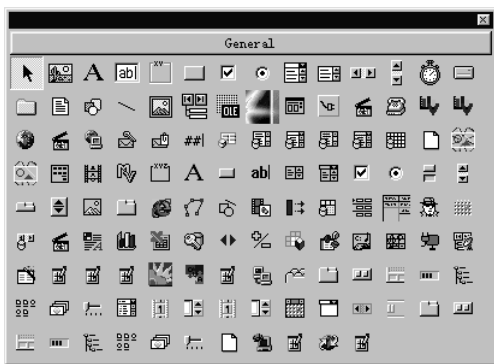


图 4-1 添加控件后的工具箱

#### 注意：

选择菜单“工程”/“部件”的意思是选择“工程”菜单上的“部件”命令，以下同。

#### 2. SSTab 控件

在 Visual Basic 6.0 所提供的控件中，有一些控件可以作为容器控件，在其上可以放置一些 VCL 控件来进行控件的窗体界面设计，如 Frame 控件和 SSTab 控件等，下面就来介绍一下 SSTab 控件。



图 4-2 在窗体上的 SSTab 控件

它的属性设置如表 4-1 所示。

表 4-1 SSTab 控件的属性设置

属 性	设 置
(Name)	SSTab1
Caption	Tab 1
Enabled	True
Height	2415
Left	240
Picture	(None)
Tab	1
TabHeight	295
Tabs	3
TabsPerRow	3
Top	120
Width	4095
WordWrap	True

说明：

控件的名称为 SSTab1，它有三个选项，即选项 0、选项 1 和选项 2，现在选中的是选项 1，现在处于有效的状态，它的显示文字为“Tab 1”，图片为空。

SSTab 控件的状态可以在程序中动态的修改，比如在窗体的设计阶段双击窗体，在窗体的 Form\_Load() 事件中添加下列代码：

```
Private Sub Form_Load()
    SSTab1.TabCaption(1) = "选项 1"
    '设置选项 1 上的文本
    SSTab1.TabVisible(2) = False
```

设置选项 2 上的可见状态

```
SSTab1.TabEnabled(0) = False
```

设置选项 0 的有效状态

```
SSTab1.Tab = 1
```

设置选项 1 为选中的状态

```
SSTab1.Tabs = 4
```

添加一个选项

```
End Sub
```

保存文件，运行程序，结果如图 4-3 所示。



图 4-3 程序运行结果

### 3. 向窗体上添加控件

向窗体上添加一个 CommandButton 控件、两个 TextBox 控件、一个 SSTab 控件、四个 ToggleButton 控件、两个 UpDown 控件和四个 Label 控件。

添加控件后的窗体如图 4-4 所示。



图 4-4 添加控件后的窗体

控件说明：

- CommandButton 控件用来结束程序的运行，为程序的正常结束提供一个出口；
- TextBox 控件用来显示程序中的两个变量 X 和 Y 的值；
- UpDown 控件用来实现变量的动态更新；
- ToggleButton 控件用来实现不同的加减乘除运算；
- Label 控件用来显示文本和运算结果；
- SSTab 控件用来显示不同的选项，同时作为 ToggleButton 控件的容器。

#### 4. 添加代码

下面以乘法操作为例来添加代码，在程序的设计阶段双击 ToggleButton3 控件，在它的 ToggleButton3\_Click() 事件中添加下列代码：

```
Private Sub ToggleButton3_Click()  
Label4.Caption = Val(Text1.Text) * Val(Text2.Text)  
乘法操作  
End Sub
```

其余控件的代码添加过程在这里就不加赘述，详细的代码请参看附后的完整源程序代码。

#### 5. 运行程序

添加完代码后，按功能键 F5 运行程序，程序运行的初始画面如图 4-5 所示。

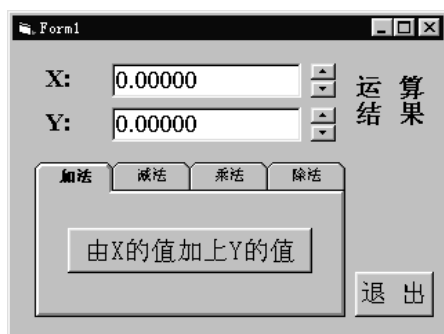


图 4-5 程序运行初始画面

在文本框中输入变量 X 和 Y 的值，选择一种算法，如除法操作，单击相应的按钮，结果如图 4-6 所示。



图 4-6 程序的运行结果

附程序的完整代码：

#### 程序清单

```
Version 6.00
```

```
Private Sub Form_Load()
```

```
    SSTab1.TabCaption(0) = "加法"
```

```
    SSTab1.TabCaption(1) = "减法"
```

```
    SSTab1.TabCaption(2) = "乘法"
```

```
    SSTab1.TabCaption(3) = "除法"
```

```
    '设置控件的显示文本
```

```
    If Text1.Text = "" Then
```

```
Text1.Text = "0.00000"
```

```
End If
```

```
'初始化文本框
```

```
If Text2.Text = "" Then
```

```
Text2.Text = "0.00000"
```

```
End If
```

```
'初始化文本框
```

```
End Sub
```

```
Private Sub Text2_Change()
```

```
If Text2.Text = 0 Then
```

```
ToggleButton4.Enabled = False
```

```
Else
```

```
ToggleButton4.Enabled = True
```

```
End If
```

```
'如果文本框 2 中的数字为 0
```

```
'那么除法无效
```

```
End Sub
```

```
Private Sub ToggleButton1_Click()
```

```
Label4.Caption = Val(Text1.Text) + Val(Text2.Text)
```

```
'加法操作
```

```
End Sub
```

```
Private Sub ToggleButton2_Click()
```

```
Label4.Caption = Val(Text1.Text) - Val(Text2.Text)
```

```
'减法操作
```

```
End Sub
```

```
Private Sub ToggleButton3_Click()
```

```
Label4.Caption = Val(Text1.Text) * Val(Text2.Text)
```

```
'乘法操作
```

```
End Sub
```

```
Private Sub ToggleButton4_Click()
```

```
Label4.Caption = Val(Text1.Text) / Val(Text2.Text)
```

```
'除法操作
```

```
End Sub
```

```
Private Sub UpDown1_DownClick()
```

```
If Text1.Text > 0 Then
```

```
Text1.Text = Text1.Text - 1
```

```
End If
```

```
'单击向下的箭头，数字减 1
```

```
End Sub
```

```
Private Sub UpDown1_UpClick()
```

```
If Text1.Text = "" Then
```

```
Text1.Text = 0
```

```
End If
```

```
Text1.Text = Text1.Text + 1
```

```
'单击向上的箭头，数字加 1
```

```
End Sub
```

```
Private Sub UpDown2_DownClick()
```

```
If Text2.Text > 0 Then
```

```
Text2.Text = Text2.Text - 1
```

```
End If
```

```
'单击向下的箭头，数字减 1
```

```
End Sub
```

```
Private Sub UpDown2_UpClick()
```

```
If Text2.Text = "" Then
```

```
Text2.Text = 0
```

```
End If
```

```
Text2.Text = Text2.Text + 1
```

单击向上的箭头，数字加 1

```
End Sub
```

## 4.2 多重窗体设计

一个大的应用程序往往不会只包括一个窗体，否则程序就显得有一些单薄，而多窗体程序设计可以使程序变的丰富多彩，在多重窗体中每个单独的窗体都有自己的特点与功能，所以程序的功能会更加强大。

### 4.2.1 多重窗体设计常用方法

在多窗体的程序设计中经常要用到下面四种方法：Load 方法、Show 方法、Hide 方法和 Unload 方法。

#### 1. Load 方法

它的语法结构如下：

```
Load [窗体名称]
```

使用 Load 方法调用的窗体只是被装入内存，并不会显示出来，与此同时，会产生一个 Form\_Load()事件。如：

```
Load Form2
```

#### 2. Show 方法

它的语法结构如下：

```
[窗体名称].Show
```

使用 Show 方法会显示被调用的窗体，如果在调用 Show 方法之前没有把窗体调入内存，那么调用 Show 方法会自动的把窗体调入内存。如：

```
Load Form2
```

```
Form2.Show
```

### 3. Hide 方法

它的语法结构如下：

```
[窗体名称].Hide
```

使用 Hide 方法会隐藏被调用的窗体，但是在调用 Hide 方法之后不会把窗体移出内存，但是被调用的窗体中的属性等已经处于无效的状态。如：

```
Form1.Hide
```

```
Form2.Show
```

### 4. Unload 方法

它的语法结构如下：

```
Unload [窗体名称]
```

使用 Unload 方法会从内存中移去被调用的窗体，与此同时，窗体中的变量和属性等都会处于无效的状态，在移去窗体的同时会产生一个 Form\_QueryUnload() 事件。如：

```
Form1.Show
```

```
Unload Form2
```

## 4.2.2 多重窗体设计实例

下面就以一个示例来说明如何进行多重窗体的程序设计，具体的设计步骤如下：

### 1. 启动新项目

首先启动一个新的项目，在屏幕上就会出现一个空白的窗体，窗体的属性设置如表 4-2 所示。

表 4-2 窗体的属性设置

属 性	设 置
(Name)	Form1
BorderStyle	3 - Fixed Dialog

续 表

属 性	设 置
Caption	Form1
Enabled	True
Height	2940
Left	0
MDIChild	False
Moveable	False
StartPosition	2 - CenterScreen
Top	0
Visible	True
Width	4065

这样设置的窗体有以下特性：

- 窗体在程序的运行过程中始终位于屏幕的中央；
- 窗体的大小不能够在程序的运行过程中改变；
- 在程序的运行过程中不能够移动窗体。

## 2. 添加控件

在空白的窗体上添加窗体如下控件：五个 Label 控件、四个 TextBox 控件和两个 CommandButton 控件。添加控件后的窗体如图 4-6 所示。



图 4-7 添加控件后的窗体

**技巧：**

添加上述几类常用控件的方法非常简单，只需要在“工具箱”上双击需要的控件即可。

其中控件的属性设置如表 4-3 所示。

表 4-3 控件的属性设置

属 性	设 置
(Name)	Label1
Caption	企业季度销售情况
(Name)	Label2
Caption	第一季度
(Name)	Label3
Caption	第二季度
(Name)	Label4
Caption	第三季度

续 表

属 性	设 置
(Name)	Label5
Caption	第四季度

(Name)	Text1
Text	
(Name)	Text1
Text	
(Name)	Text1
Text	
(Name)	Text1
Text	
(Name)	Command1
Caption	图 形
(Name)	Command2
Caption	退 出

由于篇幅有限，所以在这一代码的添加过程就不加赘述了，详细的代码请参看附后的完整程序代码。

### 3. 添加窗体

选择菜单“工程”/“添加窗体”，就会弹出如图 4-8 所示的对话框。

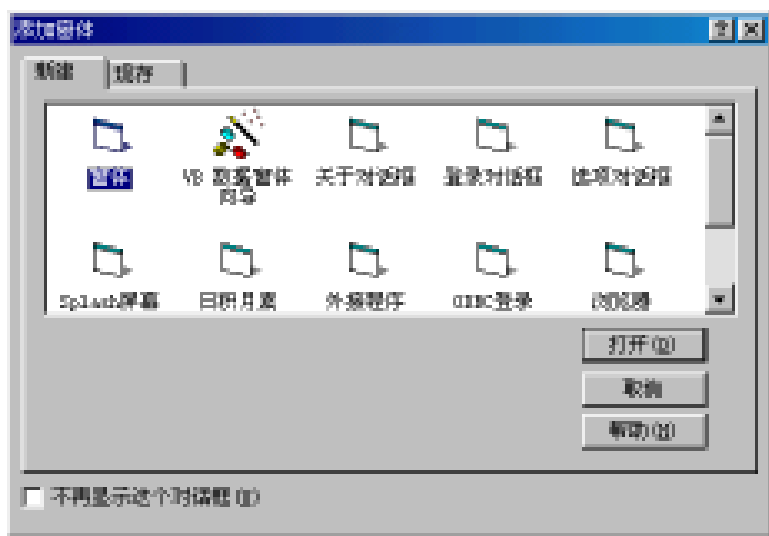


图 4-8 添加窗体对话框

在“添加窗体”对话框中选择“窗体”项，单击“打开”按钮，系统就会自动的向原有的项目中添加一个窗体。

#### 4. 添加控件

在新增的窗体上放置一个 CommandButton 控件，它的作用是在隐藏新增的窗体的同时显示原有的窗体。

添加控件后的新增窗体如图 4-9 所示。



图 4-9 添加控件后的新增窗体

窗体及控件的属性设置如表 4-4 所示。

表 4-4 窗体的属性设置

项 目	内 容
(Name)	Form2
BorderStyle	3 - Fixed Dialog
Caption	Form2
Height	2970

Left	0
Moveable	False
StartPosition	2 - CenterScreen
Top	0
Width	4275
(Name)	Command1
Caption	继续

在“继续”按钮的 Command1\_Click()事件中添加下列代码：

```
Private Sub Command1_Click()
    Form2.Hide
    '隐藏窗体 2
    Form1.Show
    '显示窗体 1
End Sub
```

在窗体 Form2 被显示的过程中，单击“继续”按钮，就会激活 Command1\_Click()事件，然后通过 Form2.Hide 和 Form1.Show 两条语句实现隐藏窗体 2 和显示窗体的功能。

#### 5. 添加说明模块

选择菜单“工程”/“添加模块”，就会弹出如图 4-10 所示的对话框。