

Visual Basic 编程基础



实验 1 窗体模块

实验目标

通过实验建立窗体模块，熟悉窗体中的“通用过程”和“事件处理过程”的区别。

实验内容

分别利用窗体中的“通用过程”和“事件处理过程”编写一个计算直角三角形边长的程序。输入两个直角边的值，单击“计算”按钮，自动计算出斜边的值。程序运行后的界面如图 3.1 所示。

实验说明

1. 线形 (line) 控件。

线形控件可以在窗体中显示一条直线，在本实验中，用三个线形控件构成了一个直角三角形。线形控件的外观如图 3.2 所示。

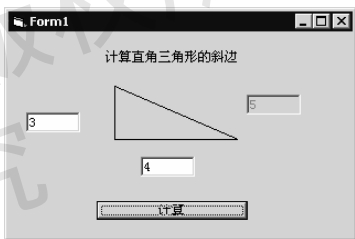


图 3.1 计算直角三角形的斜边边长

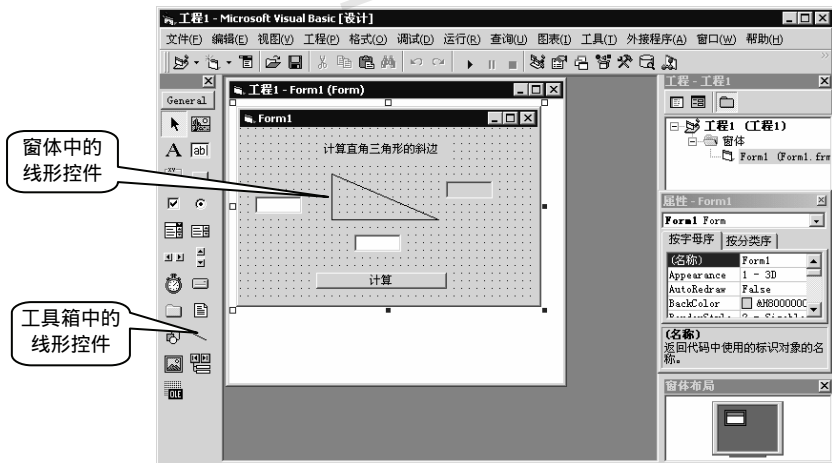


图 3.2 线形控件的外观

2. 开平方函数的使用。

直角三角形三条边之间的关系是： $A^2+B^2=C^2$ 。已知 A 、 B 两条边的值，求斜边 C 的值，需要用到开平方函数。开平方函数的使用方法为：`Sqr (number as double)`，并以双精度数格



式返回一个双精度数的平方根值。例如， $\text{Sqr}(25)=5$ 。

实验分析

通过对实训内容进行认真分析，并结合 Visual Basic 软件的功能及操作，我们可以将实验内容分解如下：

首先启动 Visual Basic，并建立用户窗体；然后加载用于显示标题“计算直角三角形的斜边”的标签控件，并设置其属性；接着分别加载用于显示三角形各条边长度值的文本框控件，并设置其属性；之后加载用于显示“计算”的命令按钮，并设置其属性；然后利用线形控件绘制一个直角三角形；最后编写代码程序。

示范操作

1. 建立窗体。

(1) 启动 Visual Basic，选择“标准 EXE”选项，进入 Visual Basic 编程环境。此时 Visual Basic 已经自动建立了一个用户窗体 Form1。

(2) 用鼠标拖动窗体边角，将窗体调整到合适的大小。

2. 加载用于显示标题“计算直角三角形的斜边”的标签控件。

(1) 加载用于显示标题“计算直角三角形的斜边”的标签控件 Label1。

(2) 通过属性窗口设置其 Caption 属性为“计算直角三角形的斜边”。

3. 加载用于显示第一条直角边的输入框的文本框控件。

(1) 加载用于显示第一条直角边的输入框的文本框控件 Text1。

(2) 通过属性窗口设置其属性如下：

Text 属性为“ ”

Name 属性为“ A ”

4. 加载用于显示第二条直角边的输入框的文本框控件。

(1) 加载用于显示第二条直角边的输入框的文本框控件 Text2。

(2) 通过属性窗口设置其属性如下：

Text 属性为“ ”

Name 属性为“ B ”

5. 加载用于显示计算出的斜边的值的文本框控件。

(1) 加载用于显示计算出的斜边的值的文本框控件 Text3。

(2) 通过属性窗口设置其属性如下：

Text 属性为“ ”

Name 属性为“ C ”

BackColor 属性为“ 灰色 ”

Enable 属性为“ False ”

6. 加载用于显示“计算”的命令按钮。

(1) 加载用于显示“计算”的命令按钮 Command1。

(2) 通过属性窗口设置其 Caption 属性为“计算”。

7. 利用线形控件在 Form1 中绘制出一个直角三角形，以突出程序的直观性。

8. 编写代码程序。

程序代码

为了突出窗体中的“通用过程”和“事件处理过程”的区别，分别按照“事件处理过程”和“通用过程”两种方法来完成这一实验。

1. 利用事件处理过程完成实验：

```
Private Sub Command1_Click()  
    '如果按下“计算”按钮，计算三角形斜边的值  
    C = Sqr(A * A + B * B)  
End Sub
```

2. 利用通用过程完成实验：

```
Private Sub js()  
    '计算三角形斜边值的通用过程  
    C = Sqr(A * A + B * B)  
End Sub  
Private Sub Command1_Click()  
    '如果按下“计算”按钮，调用通用过程 js，计算三角形斜边的值  
    js  
End Sub
```



实验 2 标准模块

实验目标

通过实验建立一个标准模块，熟悉调用标准模块的方法，比较标准模块与窗体模块的异同。

实验内容

编写一个 100 以内可以自动出题的加法练习程序。程序运行后，界面如图 3.3 所示。



图 3.3 100 以内加法练习程序

实验说明

本实验利用 Int()函数和 Rnd()函数相配合，产生 100 以内的随机整数。

Int()为取整函数，其用法为：INT(number)，例如：INT(1.5)=1。

Rnd()为随机函数，可以产生一个 0~1 的随机数字。由于 Rnd()的结果在 0 和 1 之间，所以，如果想产生一个 0~100 的随机数，应将其结果乘以 100，即 Rnd()*100，为了取得一个 100 以内的随机整数，需要利用 Int()函数为其取整，即 Int(Rnd()*100)。

实验分析

通过对实训内容进行认真分析,并结合 Visual Basic 软件的功能及操作,我们可以将实验内容分解如下:

首先建立一个窗体,然后分别加载用于显示加法运算的第一个运算数、第二个运算数、“+”、“=”、“加法运算程序”的标签控件,并设置其属性;接着加载用于输入答案的文本框控件,并设置其属性;之后加载用于出题的命令按钮控件,并设置其属性;最后编写代码程序。

示范操作

1. 建立窗体。

(1) 建立一个窗体 Form1。

(2) 将窗体调整到合适的大小。

2. 加载用于显示加法运算的第一个运算数的标签控件。

(1) 加载用于显示加法运算的第一个运算数的标签控件 Label1。

(2) 通过属性窗口设置其 Caption 属性为“ ”。

3. 加载用于显示加法运算的第二个运算数的标签控件。

(1) 加载用于显示加法运算的第二个运算数的标签控件 Label2。

(2) 通过属性窗口设置其 Caption 属性为“ ”。

4. 加载用于显示“+”号的标签控件。

(1) 加载用于显示“+”号的标签控件 Label3。

(2) 通过属性窗口设置其 Caption 属性为“+”。

5. 加载用于显示“=”号的标签控件。

(1) 加载用于显示“=”号的标签控件 Label4。

(2) 通过属性窗口设置其 Caption 属性为“=”。

6. 加载用于显示“加法运算程序”的标签控件。

(1) 加载用于显示“加法运算程序”的标签控件 Label5。

(2) 通过属性窗口设置其 Caption 属性为“加法运算程序”。

7. 加载用于输入答案的文本框控件。

(1) 加载用于输入答案的文本框控件 Text1。

(2) 通过属性窗口设置其 Text 属性为“ ”。

8. 加载用于出题的命令按钮控件。

(1) 加载用于出题的命令按钮控件 Command1。

(2) 通过属性窗口设置其 Caption 属性为“出题”。

9. 编写代码程序。

程序代码

在本实验中,为了突出窗体模块与标准模块的区别,我们分别使用窗体模块与标准模块两种方法来完成这一实验。

1. 利用窗体模块完成实验:

```
public a,b,c
'定义全局变量 a , b , c
Private Sub Command1_Click()
'如果单击“出题”按钮,则产生 100 以内的随机整数 a 和 b
```

```
a = Int(Rnd() * 100)
b = Int(Rnd() * 100)
'清空答案
Text1.Text = ""
'显示第一个随机整数
Label1.Caption = a
'显示第二个随机整数
Label2.Caption = b
End Sub
```

2. 利用标准模块完成实验：

```
Public a, b, c
'定义全局变量 a,b,c
Public Sub ct()
    '标准模块“出题”，首先产生 100 以内随机整数 a 和 b
    a = Int(Rnd() * 100)
    b = Int(Rnd() * 100)
End Sub

Private Sub Command1_Click()
    '如果单击“出题”按钮，则调用“出题”标准模块，产生两个 100 以内随机整数
    Call ct
    '清空答案
    Text1.Text = ""
    '显示第一个随机整数
    Label1.Caption = a
    '显示第二个随机整数
    Label2.Caption = b
End Sub
```



实验 3 类模块

实验目标

通过实验建立一个自定义的类模块，并用这个类模块进行编程，熟悉为自定义类模块建立方法、属性的操作方法。

实验内容

利用自定义类模块编制一个加法运算器程序，要求通过类模块实现运算的过程，类模块要具有一定的属性和方法。程序运行后的界面如图 3.4 所示。

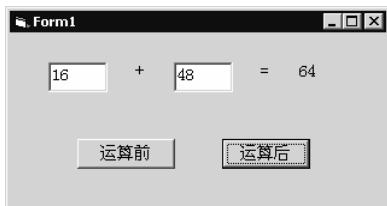


图 3.4 利用类模块建立的加法运算器

实验说明

本实验通过简单的示例来说明在 Visual Basic 中类模块的定义及其引用方法。

实验分析

通过对实训内容进行认真分析,并结合 Visual Basic 软件的功能及操作,我们可以将实验内容分解如下:

首先建立一个窗体,并调整其大小;然后编写类模块程序,并建立类模块实例;接着分别加载用于显示第一个数字和第二个数字的文本框控件,并分别设置其属性;之后分别加载用于显示“+”、“=”、显示计算的结果的标签控件,并分别设置其属性;加载用于显示结果的命令按钮控件,并设置其属性;最后编写程序代码。

示范操作

1. 建立窗体。

- (1) 建立一个窗体 Form1。
- (2) 将其调整到合适的大小。

2. 编写类模块程序。

建立类模块程序后,进行类模块程序的进一步设计时,在 Visual Basic 代码编辑器看到相应的提示,如图 3.5 所示。

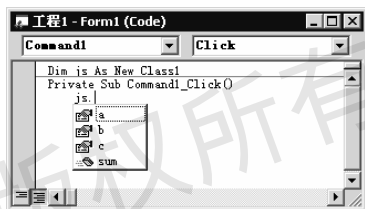


图 3.5 引用类模块编程时的提示信息

3. 建立类模块的实例。

在 Visual Basic 中,只有建立了一个类模块的实例,才能使用它。建立类模块实例的方法是:

```
类名 Dim 变量 as New
```

在本实验中,我们在建立窗体模块时首先要建立一个类模块的实例:

```
Dim js As New Class1
```

4. 加载用于显示第一个数字输入框的文本框控件。

- (1) 加载用于显示第一个数字输入框的文本框控件 Text1。
- (2) 通过属性窗口设置其 Text 属性为“ ”。

5. 加载用于显示第二个数字输入框的文本框控件。

- (1) 加载用于显示第二个数字输入框的文本框控件 Text2。
- (2) 通过属性窗口设置其 Text 属性为“ ”。

6. 加载用于显示“+”号的标签控件。

- (1) 加载用于显示“+”号的标签控件 Label1。
- (2) 通过属性窗口设置其 Caption 属性为“+”。

7. 加载用于显示“=”号的标签控件。

- (1) 加载用于显示“=”号的标签控件 Label2。
- (2) 通过属性窗口设置其 Caption 属性为“=”。

8. 加载用于显示计算结果的标签控件。

- (1) 加载用于显示计算结果的标签控件 Label3。

(2) 通过属性窗口设置其 Caption 属性为 “ ”。

9. 加载用于直接显示结果的命令按钮控件。

(1) 加载用于直接显示结果的命令按钮控件 Command1。

(2) 通过属性窗口设置其 Caption 属性为 “ 运算前 ”。

10. 加载用于用 Sum 方法进行运算后再显示结果的命令按钮控件。

(1) 加载用于用 Sum 方法进行运算后再显示结果的命令按钮控件 Command2。

(2) 通过属性窗口设置其 Caption 属性为 “ 运算后 ”。

11. 编写程序代码。



程序代码

1. 在工程中建立类模块 Class1，并编写如下代码：

```
Public a, b, c As Integer
'建立类模块的属性 a,b,c，并设置其类型为 Integer
Public Sub sum()
'建立类模块的求和方法 sum()
    c = a + b
    '进行求和运算
End Sub
```

2. 窗体代码：

```
Dim js As New Class1
'建立一个 Class1 类的实例 js
Private Sub Command1_Click()
'运算前事件处理过程
    js.a = Val(Text1.Text)
    '设置 js 的属性 a 为第一个操作数
    js.b = Val(Text2.Text)
    '设置 js 的属性 b 为第二个操作数
    Label3.Caption = js.c
    '显示结果，结果为 js 的属性 c 的取值
End Sub
```

```
Private Sub Command2_Click()
'运算后事件处理过程
    js.a = Val(Text1.Text)
    '设置 js 的属性 a 为第一个操作数
    js.b = Val(Text2.Text)
    '设置 js 的属性 b 为第二个操作数
    js.sum
    '利用 js 的 sum 方法进行运算
    Label3.Caption = js.c
    '显示结果
End Sub
```

常量和变量



实验 1 Visual Basic 中数据的使用

实验目标

学会 Visual Basic 中数据的使用方法，掌握辨析合法与非法的 Visual Basic 数据的方法，以及 Print 语句的使用方法。

实验内容

(1) 上机验证教材第 4 章 4.1 节中介绍的各种数据。

(2) 上机验证教材第 4 章后习题 4 中简答题的第(3)题，说出哪些是合法的 Visual Basic 数据，哪些是非法的 Visual Basic 数据，并分析其中的原因。

实验说明

本实验通过使用 Print 语句来说明在 Visual Basic 中各种数据的使用方法。

实验分析

通过对实训内容进行认真分析，并结合 Visual Basic 软件的功能及操作，我们可以将实验内容分解如下：

首先启动 Visual Basic，并进入 Visual Basic 的编程环境；然后在 Activate 事件过程的开始语句和结束语句之间输入语句，并运行代码编辑窗口中的程序，并观察运行结果；最后修改程序语句和数据，观察运行结果的变化。

示范操作

1. 启动 Visual Basic，并进入 Visual Basic 的编程环境。

(1) 启动 Visual Basic，选择“新建工程”窗口中“新建”选项卡中的“标准 EXE”，并单击“打开”按钮，进入 Visual Basic 窗口界面，这时在屏幕中央显示出窗体窗口。

(2) 单击菜单栏中的“视图”菜单，然后选择“代码窗口”选项，即可进入代码编辑窗口（单击工程资源管理器窗口上端的“查看代码”按钮，或直接双击窗体窗口可以更方便、快捷地进入代码编辑窗口）。

(3) 单击代码编辑窗口上面的“对象”列表框右端的向下箭头，并选择 Form，这时在代码编辑窗口中就会自动出现 Form 窗体的 Load 事件过程。

(4) 单击“过程”列表框右端的向下箭头，选择 Activate（激活）事件过程，这时在代码编辑窗口中就会出现 Form 窗体的 Activate 事件过程。

(5) 将 Form 窗体的 Load 事件过程从代码窗口中删除掉。

2. 在 Activate 事件过程的开始语句和结束语句之间输入语句，并观察运行结果。