实验二 数据类型及常用内部函数的简单应用

一、实验日的

1. 了解窗体窗口、图形框的构造及其属性、事件和方法。

2. 掌握用 VB 开发应用程序的一般步骤。

3. 掌握 VB 数据类型的基本概念。

4. 掌握各种运算符的功能及表达式的构成和求值方法。

5. 了解 VB 的标准函数,掌握常用标准函数的功能和用法。 反社版权所

6. 了解格式输出函数的功能及使用方法。

二、预备知识

1. 窗体和图形框

1) 窗体

窗体是所有控件的容器,工具箱上的工具只有放到窗体上才能成为控件对象,窗体还 可作为输出控件。窗体具有与其他标准控件相似的属性。

(1)常用属性。

Name 属性:该属性为字符型属性,是所有对象都具有的属性,是所创建对象的名称。 所有的控件在创建时由 VB 自动提供一个默认名称。对象的名称是作为对象的标识而引用 的,不会显示在窗体上。该属性只能在属性窗口中设置。

Caption 属性:该属性为字符型属性,用于在窗体题栏中显示的文本内容,该属性可以 在程序代码中设置。

ControlBox 属性:该属性为逻辑属性,确定在窗体的标题栏上是否有最大化、最小化 和关闭按钮。该属性只能在属性窗口中设置。

BorderStyle 属性:该属性确定了窗体的边框样式,其取值范围为 $0 \sim 5$,当该属性值为 0时,不显示窗体的标题栏。该属性在属性窗口中设置有效,程序代码中设置无效。

Picture 属性:设置窗体中要显示的图片,该属性可以在程序代码中设置。在窗体上显 示图片的语句为:窗体对象名.Picture = LoadPicture ("图片文件全名")。

(2)常用事件。

窗体最常用的事件有 Click、Activate 和 Load 等。

Load 事件:把窗体装入工作区时触发的事件,该事件通常用来在启动应用程序时对属 性和变量进行初始化、装载数据等。

Activate 事件:当一个窗体变成活动(当前)窗体时引发的事件。

Nisual Basic 案例教程实验指导

(3)常用方法。

方法是面向对象的,所以在调用时一定要指明对象。若不指明对象,表示为当前对象, 一般指窗体。

Print 方法。Print 方法的作用是在对象上输出文本内容。输出的开始位置是由对象的当前坐标属性(CurrentX和CurrentY)决定的,默认为对象的左上角。

格式: [对象.]Print [{Spc(n)|Tab(n)}] [表达式列表][,|;]

对象:可以是窗体、图形框或打印机。若省略了对象,为当前窗体。

● Spc(n)函数:用于输出时插入 n 个空格。

- Tab(n)函数:用于在输出表达式列表时从对象最左边开始向右移动 n 列。
- 表达式列表:要输出的数值或字符串表达式,若省略,则输出一个空行,多个表达 式之间用逗号或分号分隔,在表达式中可使用 Spc 和 Tab 函数。
- ●";"分号:按紧凑格式输出。

●"," 逗号:表示光标定位在下一个打印区的开始位置处,每隔14列为一个打印区。

一般情况下,每执行一次 Print 方法就要自动换行,为了使下一个 Print 输出仍在同一 行显示,可以在当前的 Print 方法末尾加上逗号","或分号";"。

当窗体对象的事件为 Load 时,在窗体对象上使用 Print 方法输出表达式,则要在第一个 Print 方法的前面加 "Show"(显示)方法或将窗体的 AutoRedraw 属性设置为 True。

Cls 方法。Cls 方法清除运行时在窗体或图形对象中显示的文本内容,不能清除窗体在设计时的文本内容。使用 Cls 方法后,当前坐标被设置为 0。

格式: [对象.] Cls

其中:对象为窗体或图形框,默认时为窗体。

Move 方法。Move 方法用于移动窗体或控件,并可改变其大小。

格式: [对象.] Move 左边距离 [,上边距离 [,宽度 [,高度]]]

对象:除时钟、菜单外的所有控件,省略对象时默认为当前窗体。

左边距离、上边距离、宽度、高度:数值表达式,以 Twip 为单位。如果对象是窗体,则"左边距离"和"上边距离"以屏幕左边界和上边界为准,否则以所在窗体的左边界和上边界为准,宽度和高度表示可改变对象的大小。

2)图形框

图形框大多数属性与窗体相同,图形框也支持 Print、Cls 和 Move 方法。但在调用这些方法时一定要指明是图形框对象。

2. VB 中的数据类型

VB 中常用的数据类型有数值型、字符型、布尔型、日期型和变体型。

1) 数值型(Numeric)

数值型包括:整型 Integer、长整型 Long、单精度型 Single、双精度型 Double、货币型

Currency、字节型 byte。

2)字符串数据类型(String)

(1) 变长字符串数据类型。

变长字符串数据类型是在声明字符数据类型时不指定字符长度。

```
Private S1 As String
Public SS As String
Dim S2 As String
```

(2) 定长字符串数据类型。

定长字符串数据类型是在声明字符数据类型时指定字符长度,在使用中字符串超过给 定长度,则系统自动将多余(超过)的字符截掉。若字符串少于给定长度,系统自动用空 格填满不够部分。

Private S1 As String * 18 Public SS As String * 1 Dim S2 As String * 6

3) 布尔数据类型 (Boolean 逻辑)

逻辑型数据,只有两种取值 True 或 False,逻辑型数据与数值型数据之间可以相互转换。逻辑型数据转换成整型数据时,True 转换为-1,False 转换为 0;数值型数据转换成逻辑型数据时,非零值为 True,零值为 False。

4) 日期 (Date) 数据类型

日期型数据包括日期与时间,日期和时间的表示范围:

日期: 公元 100 年 1 月 1 日到 9999 年 12 月 31 日。

时间:00:00:00到23:59:59。

任何字面上可被认做日期和时间的字符,只要用号码符(#)括起来,都可以作为日期 型数据。如#5/12/88#、#May 1,77#、#2000-4-9#、#1 May,1998#等。

当某一数值类型数据转换成日期类型数据时,小数点左边的数字代表日期部分,负数 代表的是 1899 年 12 月 30 日之前的日期,正数代表的是 1899 年 12 月 30 日以后的日期; 而小数点右边的数据代表时间部分,0 对应午夜 0:0:00,0.5 对应中午 12:00:00。

5) 变体数据类型(Variant)

它是较为特殊的一类数据类型,其数据类型不固定,而且是随运算参数的变化而变化。

3. 变量

每个变量都有一个名字和相应的数据类型,程序通过变量名来引用变量值。变量类型的种类与数据类型的种类相同。使用变量之前,一般须声明变量名及其类型。在 VB 中可以通过以下几种方式来声明变量及类型。

1)使用声明语句

格式:<u>Public | Private | Dim | Static</u> <u>变量名</u> [As Type]

该格式分为三个部分:第一个部分用来声明变量的作用范围(域);第二部分用来声明 变量名;第三部分用来声明变量的类型,它是一个可选项,选择则可以声明变量的类型, 省略该项,则所创建的变量默认为变体类型。一条声明语句可以同时声明多个变量,但每 个变量必须有自己的类型声明,类型声明不能共用。

例如:

○ Visual Basic 案例教程实验指导

Dim Dp4 As String, Na5 As Integer, Var1, Sb As String * 14

Dp4 被声明为一个变长字符串变量, Na5 被声明为一个整型变量, Sb 被声明为一个定 长字符串变量, Var1 被声明为一个变体变量。

2) 用类型说明符直接声明变量

格式: Public | Private | Dim | Static 变量名<类型符>

Dim Ch\$, S# 等价于 Dim Ch As String, S As Double

Private n1% 等价于 Private n1 As Integer

3) 隐式声明

未进行声明而直接使用的变量为隐式声明变量。所有隐式声明的变量都是 Variant (变体)类型,其作用范围仅限于变量所在的过程。

注: 若在"工具""选项"菜单项中选择了"要求变量声明",则事件过程中所使用的变量必须先声明后使用。

4. 常量

常量与变量一样,也需要用名字来标识,常量的命名约定与变量命名约定相同。

1) 常量的声明

[Public | Private] Const 常量名 [As Type] = Expression

第一部分用来说明常量的作用域(范围),省略该项则为过程级常量;第二部分 Const 关键字说明该语句为常量声明语句;第三部分是常量名;第四部分说明常量的数据类型, 若省略该项,常量的数据类型由表达式确定;第五部分为表达式,由数值常数或字符串常 数及运算符组成,但在表达式中不能使用函数。

一旦声明了常量,则在程序代码中不允许对常量(数)再次赋值。

Const Pi# = 3.1415926 '在事件过程中声明, Pi 是过程级常量, 在声明该常量的过程 中有效。

Private Const Ch1 As String = "Computer" '在代码窗口的通用/声明处声明, Ch1 是窗体级常量, 在声明该常量的窗体的所有事件过程中都有效。

Public Const Pail As Single = 3.1415926 '在添加的模块中声明, Pail 是工程级常量, 在声明该常量的工程的所有窗体的所有事件过程中都有效。

2) 常量的类型

(1)数值常量:由数字、正负号、小数点构成。

(2)字符常量:由一对双引号括起来的字母、数字和其他字符构成的字符串。

(3)日期常量:由一对"#"号所括起来的日期数据。

(4)逻辑(布尔)常量:该常量有两个值 True 和 False。

3) 系统提供的常量

VB 系统还提供了应用程序和控件定义的常量,在"对象浏览器"的 Vb 或 VBA 等对 象库中列举了 Visual Basic 的常量。例如, vbYesNoCancel、vbYesNo、vbOKOnly、vbCrLf、 vbRed、vbGreen、vbBlue 等。

5. 运算符

1)算术运算符

"()"括号、"^"乘方、"-"(负号)、"*"乘、"/"除、"\"整除、"Mod"取模(求余)、

·14 · Page

"+"加、"-"减。

当表达式中包含的操作符(运算符)不只一个时,计算顺序并不是从左到右进行,而 是遵循运算优先次序。其优先次序是:() ^ */ \ Mod + -。

2)字符运算符

字符运算符有两个:一个是"&",另一个是"+",它们都是将两个字符串拼接起来。 在字符串变量后使用运算符"&",变量与运算符"&"之间必须加一个空格。字符运算符 的应用见表 2.1。

表 2.1 字符运算表

运算表达式	运算结果	运算表达式	运算结果	运算表达式	运算结果
数字 & 数字	字符	字符 & 数字	字符	字符 & 字符	字符
数字+数字	数值	字符+数字	出错	字符+字符	字符

3) 关系运算符

关系运算用于对两个实体进行比较,比较结果是布尔(逻辑)值,即 True 和 False 之一。若关系成立,则返回 True,否则返回 False。关系运算的结果通常作为判断(条件),关系运算符有下列几种,它们的运算优先级别相同。

" = " 等于、" < " 小于、" > " 大于、" <> " 不等于、" <= " 小于等于、" >= " 大于等于。 进行关系运算时注意以下规则:

(1)关系运算符两边必须是相同的数据类型。

(2) 数值型按其大小比较。

(3)字符型按字符的 ASCII 码值从左到右一一比较,即首先比较两个字符串的第一个字符,其中 ASCII 码值大的字符串大,如果第一个字符相同,则比较第二个字符,以此类推,直到出现不同的字符为止。

(4)日期型按日期的先后进行比较。越靠近当前日期越大。

4)逻辑运算符

逻辑运算符主要用于对关系表达式进行运算,连接两个或多个关系表达式,组成一个逻辑表达式,其结果是逻辑值 True 和 False 之一。常用的逻辑运算符有 Not、And、Or,运算优先级为:Not And Or。

Not:逻辑非(取操作数相反的逻辑值);

And:逻辑与(只有两个操作数的逻辑值为"真"时,结果才为"真");

Or:逻辑或(只有两个操作数的逻辑值为"假"时,结果才为"假")。

6. 表达式

由变量、常量、函数、运算符和圆括号按一定的规则组成的字符序列称为表达式。表达式经过运算后有一个结果,运算结果的类型由数据和运算符共同决定。在使用表达式时 应注意:

1) 不同数据类型的转换

在算术运算中,如果操作数具有不同的数据精度,则 VB 规定运算结果的数据类型以 精度高的数据类型为准。

Integer < Long < Single < Double < Currency</pre>

Visual Basic 案例教程实验指导

2) 各类运算符的优先级

当一个表达式中出现了多种不同类型的运算符时,不同类型运算符优先级如下:

算术运算符 > 字符运算符 > 关系运算符 > 逻辑运算符

7. 常用函数

函数按其功能可分成数学函数、转换函数、字符串函数、日期/时间函数和格式输出函 数等。这些函数都带有一个或几个自变量(称为参数),函数对这些自变量进行运算,返回 一个结果值。函数的一般格式为:

```
<函数名>([参数表])
```

说明:若函数带有多个参数,参数间用逗号分隔。函数以表达式的形式调用。

1) 随机函数 Rnd

Rnd 函数返回 0~1 (不包括 0 和 1)之间的双精度随机数。为了在每次运行时产生不同序列的随机数,可以执行 Randomize 语句。不执行该语句,则每次运行时产生相同序列的随机数。

```
格式:Randomize [number]
```

其中, number参数是Variant或任何有效的数值表达式。

2)格式输出函数

用格式输出函数 Format 可以使数值、日期或字符串按指定格式输出,一般用于 Print 方法中。

```
格式:Format(表达式[, "格式字符串"])
```

表达式:数值表达式、日期表达式或字符表达式。

格式字符串:表示输出表达式值时所采用的输出格式。格式字符串有三类:数值格式、 日期格式和字符串格式。格式字符串必须用双引号括起来。

3) IsNumeric 函数

```
格式: IsNumeric (表达式)
```

功能:测试表达式是否为数字,当表达式为数字时函数值为 True,否则函数值为 False。 该函数常用于检测输入的合法性。

例如:x = "dsfgdsgf": y = "1234567"

```
Print IsNumeric(x), Not IsNumeric(x) '输出 False 和 True
Print IsNumeric(y), Not IsNumeric(y) '输出 True 和 False
```

4) Shell 函数

格式:Shell(命令字符串[,窗口类型])

功能:调用能在 DOS、Windows 下运行的可执行程序。

命令字符串:可以是字符变量,也可以是字符常量,命令字符串是要执行的应用程序 全名,包括盘符、路径、文件名.扩展名(扩展名为.bat、.exe、.com)。

例如:利用 Shell 函数调用 Windows 的"记事本"应用程序。程序代码为:

KK = Shell("C:\Windows\System32\notepad.exe",1)

·16 · Page

🖻 计算总分和平均分

图 2.1

Text1

或 Call Shell ("C:\Windows\System32\notepad.exe", 1)

或 Shell "C:\Windows\System32\notepad.exe", 1

三、实验案例

案例 1. 编制计算学生数学、英语、物理三门功课的总分和平均分的程序。姓名、数 学、英语、物理用文本框输入,由图形框实现如下格式的输出。平均分要求保留两位小数。 姓名:

数学:	英语:	物理:
总分:		
平均分:		

1) 案例分析

设 m、e、p 分别代表数学、英语、物理三门功课的变量,文本框输入的成绩用 Val 函数将其转换成数值再赋予变量之中(如 M = Val(txtMath.Text)),然后进行求和、求平均值的计算。输出的平均值保留两位小数,可用格式符("###.##")控制输出位数。输出的分隔线可在 Print 方法中利用 String(N,C)函数生成。如 String(30,"-")。

2)界面设计

根据题意在窗体上制作四个标签框(用于输入提示), 四个文本框(用于输入姓名和三门功课的成绩),三个命令 按钮("计算"按钮、"清除"按钮、"结束"按钮),一个 图形框(用于输出计算结果)。界面设计如图 2.1 所示。

3) 对象属性设置

对象属性设置如表 2.2 所示。

属性对象	默认控件名	Name	Caption	ToolTipText
窗体	Form1	Form1	计算总分和平均值	无
	Label1	lblName	姓名	空
行交近	Label2	lblMath	数学	空
你立住	Label3	lblEng	英语	空
	Label4	lblPhy	物理	空
	Command1	cmdCalc	计算	进行运算
命令按钮	Command2	cmdClear	清除	清除文本框和图形框内容
	Command3	cmdExit	结束	结束程序运行
	Text1	txtName	无	无
立木柜	Text2	txtMath	无	无
又坐框	Text3	txtEng	无	无
	Text4	txtPhy	无	无
图形框	Picture1	picShow	无	无

表 2.2 对象属性

计算

结束

物理

Text4

案例1界面设计

Text3

4) 对象事件过程代码

```
Private Sub Form Load()
 txtName.Text = ""
 txtMath.Text = ""
 txtEng.Text = ""
 txtPhy.Text = ""
End Sub
Private Sub cmdCalc Click()
Dim m As Single, p As Single, e As Single, s As Single, x As Single
 m = Val(TxtMath.Text)
 e = Val(TxtEng.Text)
 p = Val(TxtPhy.Text)
 s = m + e + p
 x = s / 3
 picShow.Print "姓名:"; txtName.Text
 picShow.Print String$(38, "-")
 picShow.Print "数学:"; m; Space(3); "英语:"; e; Space(3);
 picShow.Print "物理:"; p
 picShow.Print "总分:"; s
 picShow.Print "平均分:"; Format(x,
 picShow.Print String$(38, "-")
End Sub
Private Sub CmdClear Click()
 PicShow.Cls
 TxtName.Text =
 TxtMath.Text =
 TxtEng.Text = ""
 TxtPhy.Text = ""
 TxtName.SetFocus
End Sub
Private Sub cmdExit
                    Cl
 End
End Sub
```

5)运行界面

运行界面如图 2.2 所示。

案例 2. 编制一个允许对文本框中的文本进行剪切、 复制和粘贴的简单编辑程序。

1) 案例分析

要实现上述要求,在窗体上制作一个文本框(Name 属性值为 txtEdit),且文本框的 Multiline 属性为 True,还 要制作对文本框中的文本进行相应操作的命令按钮。

弯 计算总	分和平均	纷	I	- 🗆 ×
姓名	数学	英语	物理	
黄河水	84	67	78	计算
肺をまた	1-12	·		
姓名・東西				清除
数学: 84	英语	: 67	物理:	
□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□	5.33			结束

"复制"、"剪切"首先要在文本框中选择文本,利用文本框的 SelText 属性,获取当前 在文本框中所选择的文本,再进行复制和剪切。

剪切:把所选择的文本暂存(赋给一个窗体级变量)后,将其删除。程序代码为:

```
St = txtEdit.SelText
txtEdit.SelText = ""
```

复制:把所选择的文本暂存后,原文本的内容保持不变。程序代码为:

St = txtEdit.SelText

图 2.2 案例 1 运行界面

粘贴:将剪切或复制的文本内容,插入到当前焦点(光标)位置。实现的方法是:在 正文中选择插入点,此时将插入点当做所选择的文本,把剪切或复制的字符串插入即可。

2) 界面设计

根据题意在窗体上建立一个文本框对象,四个命令按钮对象(剪切、复制、粘贴、结束)。界面设计如图 2.3 所示。

3) 对象属性设置

对象属性设置如表 2.3 所示。

表 2.3 对象属性

属性对象	默认控件名	Name	Text	Caption	ToolTipText
窗体	Form1	Form1	无	文本编辑	无
今天持行	Command1	cmdCut	无	剪切	剪切所选文本内容
	Command2	cmdCopy	无	复制	复制所选文本内容
דזאנ א ווו	Command3	cmdPaste	无	粘贴	粘贴到指定位置
	Command4	cmdEnd	无	结束	结束程序运行
文本框	Text1	txtEdit	空	无	无

4) 对象事件代码

```
Private St As String '声明 St 为窗体/模块级变量,存放剪切或复制的字符串
Private Sub cmdCut_Click()
St = txtEdit.SelText
txtEdit.SelText = ""
End Sub
Private Sub cmdCopy_Click()
St = txtEdit.SelText
End Sub
Private Sub cmdPaste_Click()
txtEdit.SelText = St
End Sub
Private Sub cmdEnd_Click()
End
End Sub
```

5)运行界面

运行界面如图 2.4 所示。

■. 文本領	毎 ■□×
	复制
	米占贝占
	结束
图 2.3	案例2界面设计



四、实验内容

1. 打开实验一的"计算器 1",将其窗体文件和工程文件另存。打开另存文件,按图

page · 19 ·

Visual Basic 案例教程实验指与

2.5 修改其界面。运行时用数字、小数点和符号按钮输入数据并显示在文本框中,单击"计 算器 2"上的函数按钮,在文本框中显示函数计算结果。编写事件过程及代码实现"计算器 2"中各按钮的功能。

提示:""按钮是删除文本框最右边的一个字符,其代码如下:

Text1 = Left(Text1, Len(Text1) - 1)

2. 在文本框中输入一个三位数的整数,单击图形框,将这个数的百位、十位和个位分 离成三个数,在图形框中显示输入的数、分离出来的三个数及三个数的平方和。

3. 界面设计如图 2.6 所示,单击"计算"按钮,计算两个数的除法(取两位小数) 整除和求余。单击"清除"按钮,清除所有文本框的内容,并使焦点回到 Text1 文本框。 单击"结束"按钮,结束程序运行。

气计算	[器2			_	
					Text1
7	8	9	Sqr	Sin	Abs
4	5	6	Exp	Cos	Rnd
1	2	3	Fix	Int	Chr
0		+/-	CE	•	-

图 2.5 实验 1 界面设计

ii, Form1		-	
i 被除数 [Te	xt1	除数 Tes	«t2
□ 除法 /	Text3		计算
∷整除ヽ	Text4		 清除
求余 Mod	Text5		结束

图 2.6 实验 3 界面设计

4.编写程序,在图形框中输出指定范围内(范围由文本框输入)的3个随机整数和这 三个随机整数和的平方根,平方根保留四位小数。运行界面

如图 2.7 所示。

5.字符串函数的应用,在文本框中输入一个不少于 25 个字符的主字符串,由另一文本框输入一个子字符串,删除 主字符串和子字符串两端的空格。单击图形框将主字符串中 出现的第一个子字符串删除,生成一个新字符串,在图形框 中输出,运行界面如图 2.8 所示。

🐂 Form1		_ 🗆 ×
整数下界 60	整数上界	90
三个随机	【数及和的平方	根
81 三个数和的平:	76 方根=15.2971	77
计算	清除	结束

提示:

(1)子字符串一定要被主串包含,否则运行时出错。

(2)可以在图形框的单击事件代码中使用 On Error Resume Next 作为第一个语句,这样即使子字符串没有被主字符串包含也不会产生运行错误。

6. 在文本框中输入若干字符, 对输入的字符随机放大或缩小, 运行界面如图 2.9 所示。

ii, For	•1 E	X
输入字 符串	QAZWSXEDCgh4589sdMJUNHY	*
输入子	¹ 串 [gh4589sd]	
输出 结果	QAZWSXEDCMJUNHY	

图 2.8 实验 5 运行界面

🖷, Form1	
輸入支本内容 对文本进行放▲ 大、缩小或还	放大 缩小
	还原

图 2.9 实验 6 运行界面

·20 · Page

图 2.7 实验 4 运行界面

提示:

(1)单击"放大"按钮,将文本框中的字符随机放大 $2 \sim 4$ 倍,放大倍数通过随机函数 产生。放大倍数表达式为:Text1.FontSize * Int (Rnd * 3 + 2)。

(2)为了使每次运行时产生不同的随机放大倍数,程序运行时先执行 Randomize 语句。

(3)同样,单击"缩小"按钮,将文本框中的字符随机缩小1~1/3。缩小倍数表达式为:Text1.FontSize / Int (Rnd * 3 + 2)。

(4)单击"还原"按钮,字体大小恢复成初始状态(字体为9磅),此时,"放大"、"缩小"按钮变成可操作状态。

(5)不能连续进行放大或缩小操作。执行放大后,"放大"按钮呈暗淡色(不可操作), "缩小"、"还原"按钮有效。同样执行了缩小操作后,"缩小"按钮不可操作,而"放大"、 "还原"按钮有效。

7. 在文本框中输入一个数字串,将其转换成数值,用数值格式化符"#"、"0"、"."、 ","、"%"、"\$"、"E+"在图形框中使用 Format 函数显示该数,运行界面如图 2.10 所示。

8.利用 Shell 函数调用相应的应用程序,单击"记事本"按钮图标,调用记事本应用 程序;单击"画图"按钮图标,调用画图应用程序;单击"计算器"按钮图标,调用计算 器应用程序。设计界面如图 2.11 所示。



提示:

(1)在 Windows 操作系统中,"记事本"、"画图"、"计算器"是系统应用程序,其文件名分别为(NotePad.exe、Mspaint.exe、Calc.exe)。

(2)可以通过选择"开始""搜索"菜单项,查找文件。找到后右键弹出快捷菜单, 执行快捷菜单的"属性"项,在属性对话框中将该文件的路径复制下来用于 Shell 函数中。

(3)也可以通过选择"开始""程序""附件"菜单项,在"附件"中选择对应 的项,单击鼠标右键弹出快捷菜单,执行快捷菜单的"属性"项,在属性对话框中将该文 件的路径复制下来用于 Shell 函数中。

(4)图标可以自行选择,也可以自己制作。"记事本"的图标可通过选择"开始""程序""附件""记事本"菜单项,单击鼠标右键,在快捷菜单中选择"属性",在属性对话框中单击"更改图标";通过按组合键 Alt+PrintScreen 将其复制到剪贴板;打开画图程序,粘贴剪贴板内容,截取所需的图标,以位图文件保存到磁盘;最后对命令按钮的 Style和 Picture 属性进行设置,装入对应的图形文件。