

第 6 章 函数、变量在课件中的应用

学习目标

- 计算图标的属性设置。
- 变量的基本概念及运用。
- 函数在课件中的运用。

计算图标的主要作用是存放变量、函数、表达式，还可以输入复杂的脚本语言进行程序的判断和跳转，Authorware 7.0 提供的编程语言和其他语言相同，有自己的变量和函数。变量和函数是由 Authorware 的系统封装的。除了提供大量的系统变量、系统函数外，还可以允许用户自定义变量和函数。通过本章的学习，可以提高 Authorware 的设计水平。

6.1 计算图标

6.1.1 计算图标概述

打开 Authorware，在流程线上添加一个计算图标并右击，打开计算图标的属性对话框，如图 6-1 所示，通过该对话框可以实现对计算图标的属性设置。

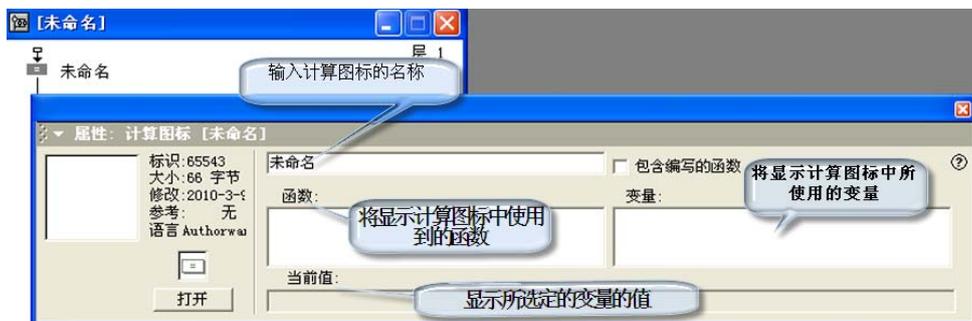


图 6-1 计算图标属性对话框各项定义

双击计算图标，打开计算图标窗口，在其中可以输入所需要的函数、变量、表达式，如图 6-2 所示。

6.1.2 计算图标应用实例

【实例描述】

在显示图标上附加一个计算图标。

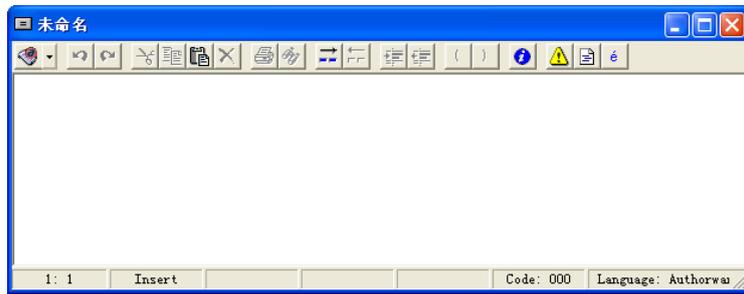


图 6-2 计算图标输入窗口

【操作过程】

(1) 在流程线上添加一个显示图标。

(2) 选中该显示图标，执行“修改”→“图标”→“计算”命令，打开计算图标窗口，并在其中输入注释内容，如图 6-3 所示。

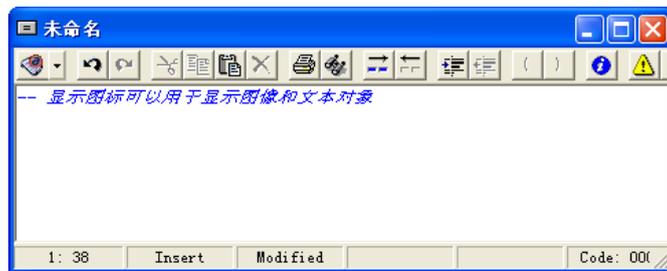


图 6-3 输入注释内容

(3) 关闭计算窗口，按照提示保存内容，则在显示图标上附加了一个计算图标，在该显示图标右上角增加了一个计算图标的标志，如图 6-4 所示。

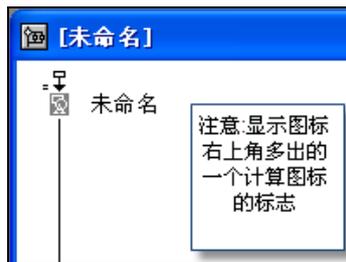


图 6-4 附加计算图标

6.2 变量在课件中的应用

6.2.1 变量简介

变量就是指能够改变的量。这些可变的量需要计算机的一些临时空间来存放。这个存储

空间就是内存。内存中不同的位置可以通过唯一的地址来识别。直接访问这些内存地址很烦琐，可以通过变量来对内存地址进行访问。

根据变量的不同来源，从使用者角度出发，Authorware 的变量可分为系统变量和自定义变量两种。

从变量中保存的数据类型出发，又可分为数值型、字符型、列表型、符号型、逻辑型、矩形变量、坐标型变量 7 种。

系统变量是 Authorware 系统定义好的变量，具有固定的变量名，用户不能更改。系统变量名一般由若干英文单词组成，大写字母开头。系统变量存放特定的参数，实现特殊功能，使用者可以在设计中直接调用。Authorware 提供的系统变量分为计算机管理教学（CMI）、决策（Decision）、文件管理（File）等 11 类。单击“窗口”→“面板”→“变量”命令，弹出如图 6-5 所示的“变量”对话框。



图 6-5 “变量”对话框

用户为了某种需要在程序中自己定义的变量称为自定义变量，变量名必须以字母开头，可以包括字母、数字、下划线，如 A1、bc1_、xy_7 等。自定义变量第一次使用时必须先赋值，默认值为 0。

变量在以下图标及对话框中使用：

- 计算图标
- 显示图标窗口中用文本工具在大括号中书写变量名称，显示变量值
- 在图标属性对话框的文本栏中填入变量名，用作对象参数

系统变量的使用方法：打开图标或图标属性对话框，单击“窗口”按钮，选择“变量”选项，在弹出的“变量”对话框中选择变量类别或用英文字母排列顺序选择变量，单击“粘贴”按钮，再单击“完成”按钮。

6.2.2 常用变量应用实例

【实例描述】

在显示图标中使用系统变量。

【操作过程】

- (1) 新建一个文件，执行“文件”→“保存”命令，将新文件命名为“系统变量演示”

并保存。

(2) 在流程线上添加一个显示图标，命名为“系统变量的使用”，双击该显示图标，在显示图标中输入如图 6-6 所示的文本内容。

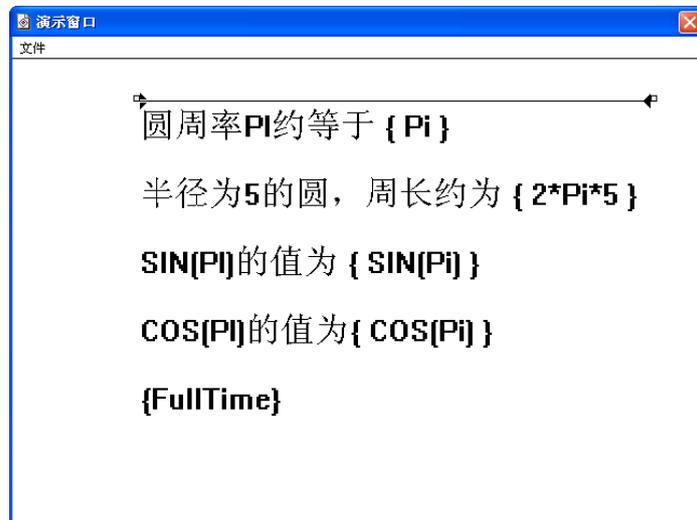


图 6-6 显示图标内容

(3) 运行程序，效果图如图 6-7 所示。



图 6-7 初始效果

(4) 要在文本对象中嵌入变量、函数和表达式，此时必须手工输入花括号，注意花括号不能嵌套使用。

(5) 在默认情况下，程序只在运行到某个设计图标时才对该设计图标中包含的文本对象进行计算和显示。如果文本对象中内嵌的变量的值或表达式的计算结果在后来发生变化，那么除非程序再次执行该设计图标，否则程序演示窗口中依然显示先前的内容。通过打开设计图标属性面板中的“更新显示变量”复选框，可以使程序在运行期间自动根据嵌入到文本对象中的

变量的当前值更新文本对象的显示结果。这里有一个最典型的例子：通过在文本对象中嵌入反映系统当前时间的系统变量 FullTime，并设置对应设计图标的“更新显示变量”属性，可以实现在演示窗中实时显示当前时间的功能，如图 6-8 所示。

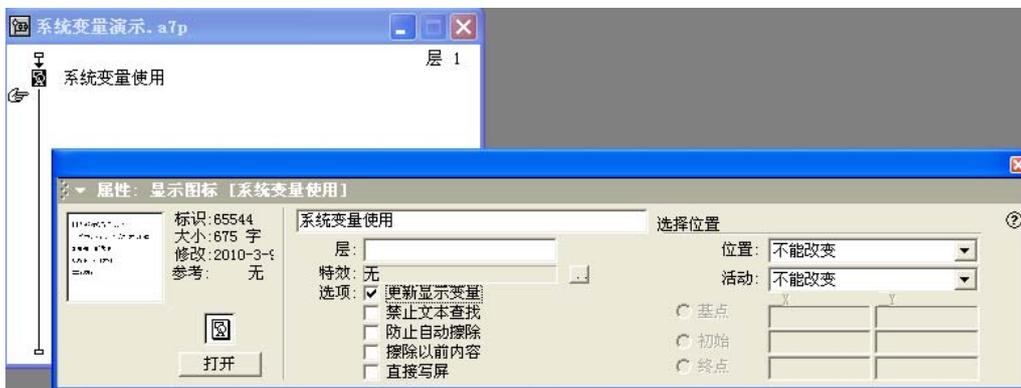


图 6-8 系统变量属性对话框

实训项目五 课件中变量的使用

【实训学时】

1 课时

【实训目的】

- 掌握常用变量的用法。
- 在课件中使用变量能增加课件的交互性，在一些物理、数学教学课件中比较常见。

【实训任务】

当程序运行时，如果按下键盘上的某一个键，屏幕上就会显示该键的名称。

【实训操作步骤】

(1) 可以创建如图 6-9 所示的流程图，在程序的开始放置一个“等待”图标，在等待图标属性对话框中选择“按任意键”复选框。在程序的最后，要设置变量和函数跳转功能，此功能是为了能不断地进行键盘输入识别。GoTo(IconID@"按钮")函数的功能在前面已经多次提到，它会使程序从后面跳到“按钮”图标处继续执行。

(2) 在流程线上打开“显示键名”显示图标，在演示图标的文本框中输入如图 6-10 所示的文字。在 Key 变量的两侧同样要加上一对双括号“{ }”。当程序运行时，按下键盘上的删除键后，屏幕上就会出现 Backspace 字样，如图 6-11 中的箭头所指。

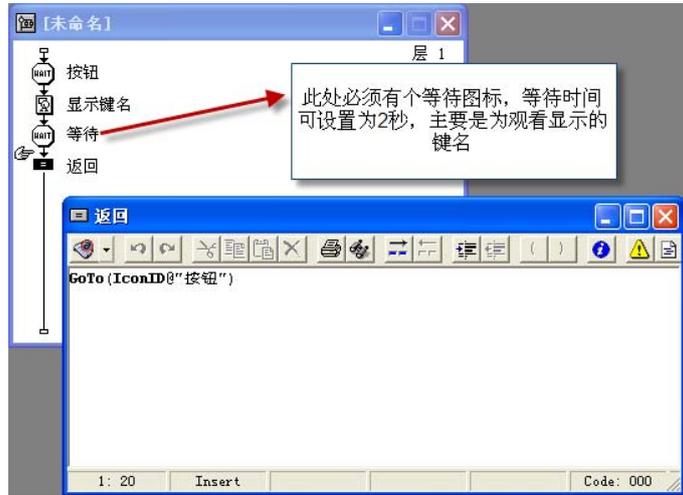


图 6-9 设置变量和函数跳转功能



图 6-10 输入变量 Key



图 6-11 按下“向上”键时

(3) Key 变量常常用于存储用户最后一次按键的键名，键名包括 a、X 等字母键，Enter、+、Esc、F2、F4、Home、Tab 等功能键和 UpArrow 等方向键。另外，在此类变量中还有一个与它功能相似的 KeyNum 变量，不过该变量只能用于存储用户最后一次按键的数字键。注意 Key 变量对 F1 键不能进行识别，因为 F1 通常被设为提示帮助的快捷键。

6.3 函数在课件中的应用

6.3.1 函数简介

在 Authorware 7.0 中函数的主要作用是对变量或者对程序的运行进行控制。和变量相类似，Authorware 7.0 中可以使用的函数分为两类：系统函数和自定义函数。

系统函数是 Authorware 7.0 自身提供的一整套函数。这些函数可以对图形对象、图标、网络、文件等内容进行操作。

一个函数在使用时一般都会使用到参数，参数是给函数进行处理的数据，是完成任务所必需的信息。系统函数以大写字母开头，由一个或多个单词组成。在使用函数时，函数中的一些参数不一定会全用到，而是根据实际情况用到其中的一部分。

单击工具栏中的  按钮，弹出“函数”对话框，如图 6-12 所示。Authorware 提供了多种类型的系统函数。

单击“分类”下拉列表框，如图 6-13 所示，其中显示了所有 Authorware 7.0 提供的系统函数。



图 6-12 “函数”对话框



图 6-13 “分类”下拉列表框

6.3.2 常用函数应用实例

【实例描述】

在显示图标中使用系统函数。

【操作过程】

(1) Beep()函数示例，General 类型。

Beep()函数可以控制 PC 喇叭，使系统响铃。在下面的例子中，使用 Circle 函数来绘制一组圆，Circle 函数的使用格式为 Circle(Pensize,x1,y1,x2,y2)，即在一个方框内绘制一个圆，该方框左上角的坐标是(x1,y1)，右下角的坐标为(x2,y2)。当 Pensize<0 时，圆内以黑色填充；当 Pensize = 0 时，整个圆以白色填充；当 Pensize>0 时，圆周的线条宽度将等于 Pensize 指定的像素点值，中间不填充。现在进行如下控制，即每当画完一个圆后，系统的 PC 喇叭就会响一下。在程序中使用了 repeat while 函数来进行循环控制，用户可以在“计算”图标文本框中输入如下语法结构：

```
repeat while x1<640
Circle(4, x1, y1, x2, y2)
x1 := x1 + 10
x2 := x2 + 10
y1 := y1 + 10
y2 := y2 + 10
Beep()
end repeat
```

程序运行时，每当画完一个圆后，PC 喇叭就响一下，每当循环一次，各参数就增加 10，当 X1 的值大于 640 后，程序将跳出循环。

(2) GoTo 函数示例，Jump 类型。

在前面已经多次使用过该函数，它的使用格式为 GoTo(IconID@"IconTitle"),IconID@"IconTitle"变量会获取 IconTitle (图标) 的 ID 号，因此 GoTo 函数就能使 Authorware 跳转到由此 ID 号指定的图标中去。

在跳转函数中还有一个函数 JumpFile，该函数的使用格式为：JumpFile("filename",["variable1, variable2, ...", "folder"])，其中 filename 为要打开的文件名称，variable1, variable2, ... 是该函数使用的各种参数，当函数跳转到指定文件后，它会将这些参数的值传递过去。folder 用来指定 Authorware 记录文件的目录路径，通常情况下如果不进行设置，默认路径为 C:\WINDOWS\A5WDATA\。

注意：在 Authorware 中，路径符号为“\\”，而不是通常所使用的“\”，例如 c:\suo\yu\通常要写成 c:\\suo\\yu\\。

6.4 运算符和语句在课件中的应用

6.4.1 运算符和语句简介

运算符是执行某项操作的功能符号，在 Authorware 中可以用到的运算符共有 5 个种类。

(1) 赋值运算符。

在 Authorware 中只有一个赋值运算符“:=”，它的含义是将运算符右边的值赋给运算符左

边的变量。

(2) 关系运算符。

关系运算符用来对两个对象进行比较，并产生一个逻辑结果。关系运算符可以比较的对象包括数字、字符串和字符串型的变量，如表 6-1 所示。

表 6-1 关系运算符

运算符	优先级	用法	含义	结合方向
>	6	op1>op2	大于	左到右
<	6	op1<op2	小于	左到右
>=	6	op1>=op2	大于等于	左到右
<=	6	op1<=op2	小于等于	左到右
==	7	op1==op2	等于	左到右
!=	7	op1!=op2	不等于	左到右

6.4.2 运算符和语句应用实例

【实例描述】

在下面的例子中，使用 `Box()` 函数来绘制矩形。当程序开始执行时，首先在画面上显示图 6-14 中箭头所指的提示信息。用户可以在画面上的光标闪烁处输入矩形的宽度、长度和边框宽度值。当按下回车键时，画面上又会提示使用鼠标在屏幕上单击，单击一下后，屏幕上就显示出你想要的矩形。

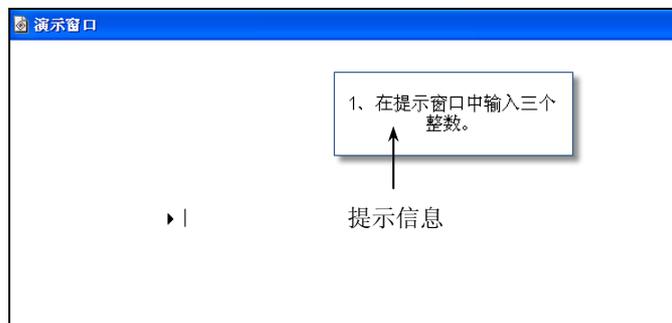


图 6-14 程序界面

【操作过程】

(1) 由于程序中要求数值输入，因此要使用文本输入响应交互，同时还需要将分支类型设为 `Try Again`，这样可以不断进行输入。打开“群组”图标，组建如图 6-15 所示的流程线。

(2) 在“提示信息”图标中输入提示信息“请在屏幕上单击绘制矩形的位置”，然后将下面的“等待”图标中设置为 `Mouse Click`，即当在屏幕上单击后程序开始往下执行，“擦除”图标将前面的提示信息擦掉，“绘图”计算图标开始绘图。打开“绘图”计算图标，在文本框中输入如图 6-16 所示的表达式。



图 6-15 编辑“群组”图标

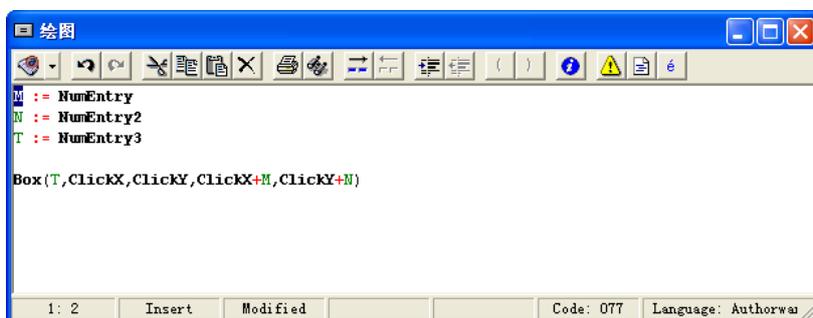


图 6-16 编辑“绘图”计算图标

(3) NumEntry、NumEntry2 和 NumEntry3 三个变量分别能够获取键盘输入的宽度、长度和边框宽度值。然后使用变量 M、N 和 T 将数值代入 Box 函数中。Box 函数的使用格式为 Box(pensize, x1, y1, x2, y2)，即用 Pensize 指定的线宽在屏幕上从(x1,y1)点到(x2,y2)点画一个方框，其中(x1,y1)是方框左上角的坐标值，(x2,y2)是方框右下角的坐标值。当 Pensize<0 时，该方框将以黑色填充；当 Pensize=0 时，该方框将以白色填充；当 Pensize>0 时，将以 Pensize 指定值的像素点个数绘制框线。在这个例子中，ClickX、ClickY 分别代表矩形的左上角位置，ClickX+M 和 ClickY+N 则代表右下角位置。

执行效果如图 6-17 所示。

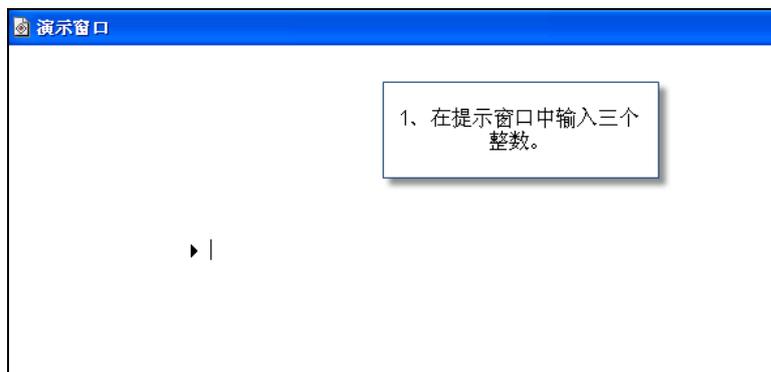


图 6-17 执行效果

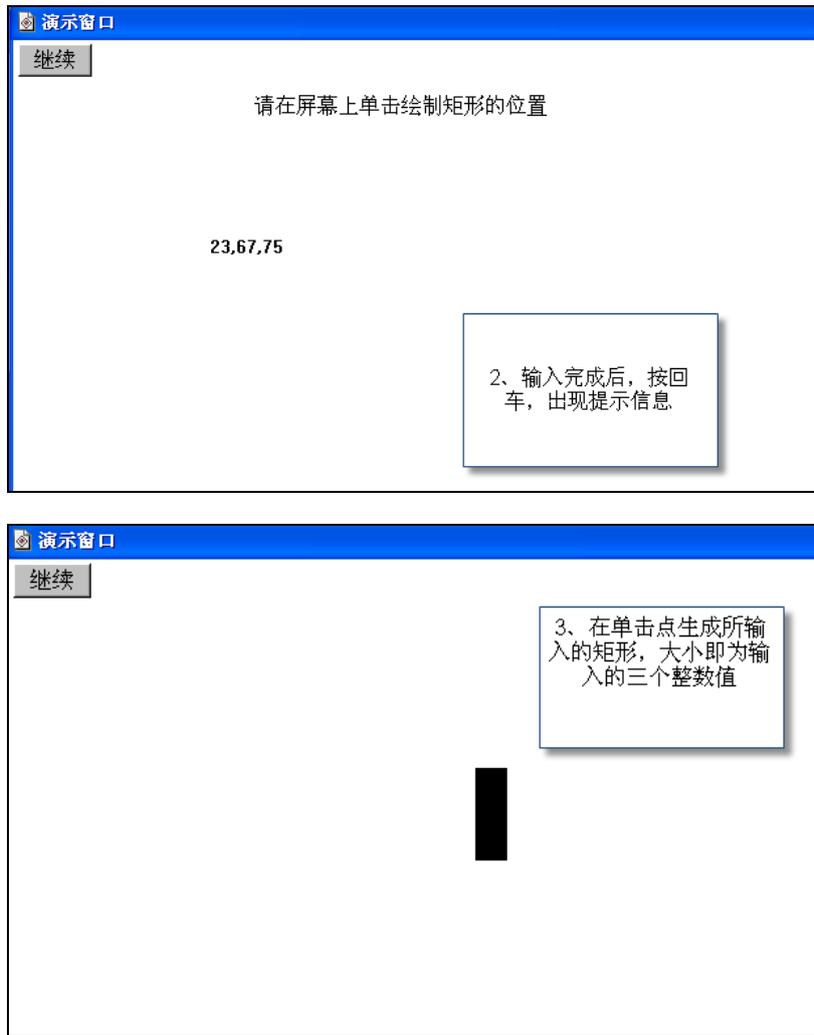


图 6-17 执行效果（续图）

实训项目六 在课件中使用函数

【实训学时】

1 课时

【实训目的】

熟悉变量和函数的综合使用。

【实训任务】

利用 Authorware 计算图标判断是否闰年。

【实训操作步骤】

(1) 新建一个项目文件，设置文件属性，然后以“闰年平年”为文件名存盘，如图 6-18 所示。



图 6-18 新建项目文件

(2) 在流程线上添加“设置窗口”计算图标，双击打开它的计算窗口，输入 `ResizeWindow(400,300)`，然后关闭计算窗口，如图 6-19 所示。

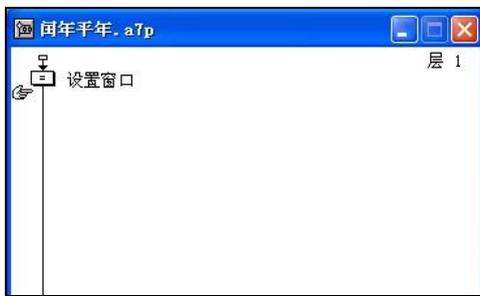


图 6-19 添加“设置窗口”计算图标

(3) 添加“输入年份”交互图标，在其属性面板的“交互作用”选项卡中选择“不擦除”，在“显示”选项卡中选中“更新显示变量”复选框，如图 6-20 所示。

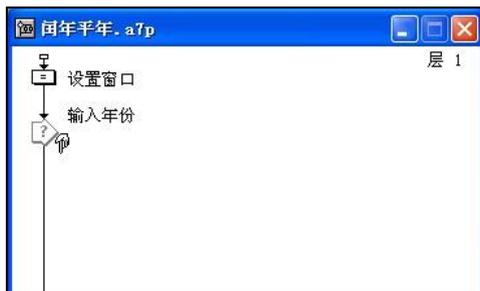


图 6-20 添加“输入年份”交互图标

(4) 双击打开交互图标（此前演示课件，然后关闭演示窗口），输入提示信息，并修改文本属性；两个新建变量的初始值分别为 0 和“闰年”（公元 0 年是闰年），如图 6-21 所示。

(5) 在交互图标右侧添加一个计算图标，选择“文本输入”交互类型，该计算图标以通配符*命名。单击“文本输入”标签，在属性面板的“响应”选项卡中选择“退出交互”分支，如图 6-22 所示。



图 6-21 修改文本属性

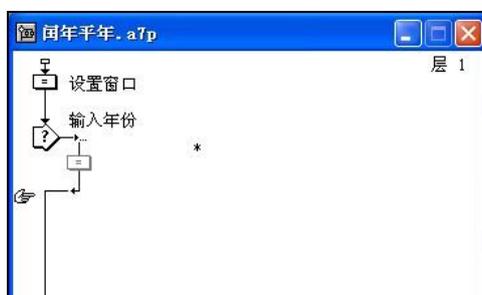


图 6-22 添加计算图标

(6) 双击打开交互图标，调整文本输入虚线框的位置和大小。双击虚线框，在“版面布局”选项卡中的“字符限制”文本框中输入4，即最多输入4位数字；在“交互作用”选项卡中清除对“退出时擦除输入的内容”复选框的选择；在“文本”选项卡中修改文本属性，如图6-23所示。

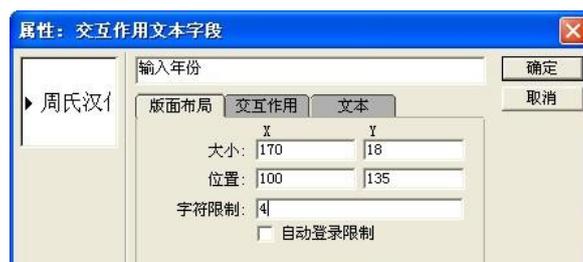


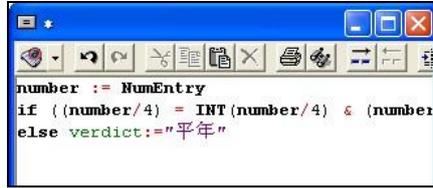
图 6-23 “版面布局”选项卡

(7) 双击打开*计算图标的计算窗口，输入以下代码：

```
number := NumEntry
if ((number/4) = INT(number/4) & (number/100) <> INT(number/100)) | (number/400) =
INT(number/400) then verdict:="闰年"
else verdict:="平年"
```

说明：第一行代码，变量 number 记录键盘输入的数据；第二行代码，判断闰年。INT 是

取整数函数，=判断相等，<>判断不等，&即 and，|即 or，如图 6-24 所示。



```
number := NumEntry
if ((number/4) = INT(number/4) & (number
else verdict:="平年"
```

图 6-24 输入代码

(8) 添加一个等待图标，修改属性，如图 6-25 所示。



图 6-25 添加等待图标

(9) 添加一个计算图标，在计算窗口中输入函数 Restart(), 用于重新计算是否闰年，如图 6-26 所示。



图 6-26 添加计算图标

(10) 打包，演示效果如图 6-27 所示。



图 6-27 演示效果

本章小结

在本章中主要介绍有关变量和表达式的知识，另外还要熟悉运算符的概念。在制作教学课件中，适当加入函数或变量可以控制教学节奏，提高交互性。比如控制视频播放的函数 `MediaPlay` 和 `MediaPause`。进行跳转的函数 `GoTo(IconID@"IconTitle")`，`IconID@"IconTitle"` 变量会获取 `IconTitle`（图标）的 ID 号，因此 `GoTo` 函数就能使 Authorware 跳转到由此 ID 号指定的图标中去。

练习题

一、选择题

- () 指的就是能够改变的量。
 - 变量
 - 函数
 - 知识对象
 - 表达式
- () 是 Authorware 7.0 自身提供的一套函数。这些函数可以对图形对象、图标、网络、文件等内容进行操作。
 - 自定义函数
 - 系统函数
 - 数学函数
 - 文字函数
- 在计算图标窗口中的引号必须在 () 状态下输入。
 - 英文
 - 全角
 - 半角
 - 中文
- 在计算图标中用户输入 `Quit()` 的意思是 ()。
 - 退出程序
 - 退出显示窗口
 - 程序无限制循环播放下去
 - 退出 Authorware
- 在计算图标窗口中用户输入 “--” 则表示 ()。
 - 可以输入程序的说明内容
 - 在程序运行时不会运行
 - 该文字没有作用
 - 它将影响程序运行

二、填空题

- 程序的脚本语句是由_____、_____、_____和_____共同构成的。
- 一个_____就是一条语句，它用来执行语句中的运算或命令等。它主要是由函数、变量、运算符构成的。
- 计算图标主要用于存放_____、_____和_____。
- 根据运算符的类型，表达式也可以分为_____、_____、_____、_____、_____等几种类型。
- 循环语句内的任何地方都可以使用_____语句提前结束本次循环，直接进入下一个循环；使用_____直接退出当前循环语句。

知识拓展

Authorware 中一些常用的判断和循环结构

(1) If...Then 结构。

这是一种条件结构，它通常用在某些事件或结果的判断上。在日常生活中，常常说“如果今天下雪，那我将去滑雪，否则的话，只有去打扫卫生了”。你可以使用类似的语言，在 Authorware 中还允许使用一个或多个条件。条件语句的使用格式如下：

```
If 条件 1 then 执行语句 1
    else if 条件 2 then 执行语句 2
else 执行语句 3
end if
```

在这个格式里，它的含义是：如果满足条件 1，Authorware 将执行语句 1；而如果满足条件 2，将执行语句 2；否则的话，Authorware 只能执行语句 3；执行完这个条件结构后，Authorware 将由 End If 来结束整个判断。

(2) Repeat With In 结构。

这是一种循环结构，通常它被应用在数组的计算上，有时也可以用来重复执行某个语句或操作，基本语法如下：

```
repeat with 变量 in 列表
    执行语句
end repeat
```

上面语法的含义是：如果变量元素在指定的列表中，该结构将重复执行下面的语句，执行完一个变量后，该函数就会指定下一个变量元素，直到该变量元素超出列表范围才由 endrepeat 来结束循环。可以阅读下面的函数示例：

```
List := [20, 30, 50]
Total := 0
repeat with X in List
    Total := Total + X
end repeat
```

在上面的例子中，指定一个数组 List，它里面有 3 个元素 20、30 和 50，下面的循环结构用来累加数组元素的和。例如，使用 X 来指定列表中的元素，当循环开始后，X 指定了列表中的第一个值 20，然后进行累加 Total= 0 + 20 = 20。在进行第二次循环后，X 被指定为 30，累加后 Total= 20 + 30 = 50。当计算完最后一个值 50 后，将退出循环，Total 的值就变为 100。