

第9章 程序调试与公布发行

- 本章目的：通过本章学习了解和掌握有关VB 程序调试、发布应用程序的方法。

第9章 程序调试与出错处理

9.1 错误类型和程序模式

9.2 程序调试

9.3 捕获错误及处理

9.3.1 Err对象

9.3.2 捕获错误

9.3.3 设计错误处理程序

9.4 程序发布

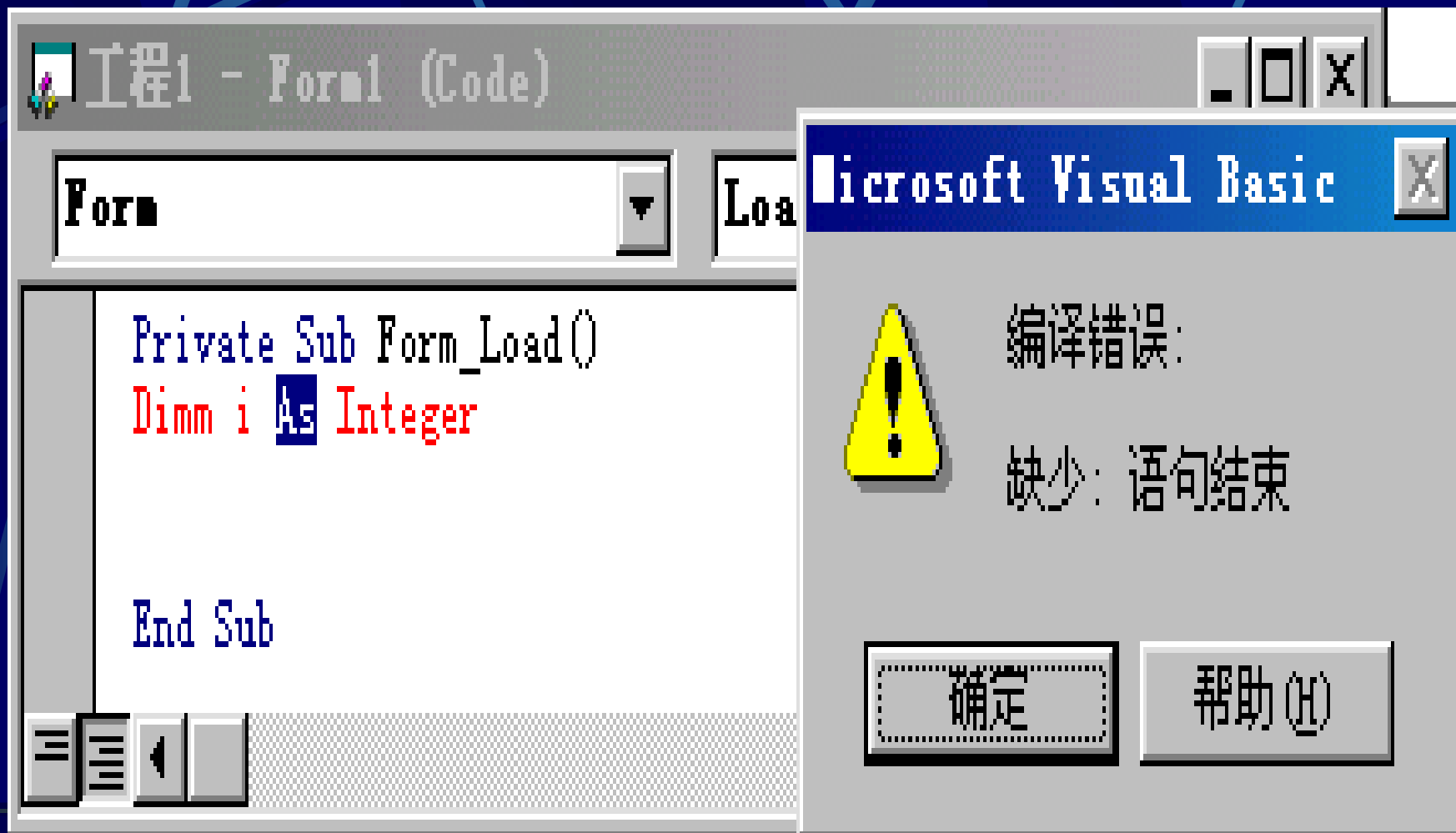
9.1 错误类型和程序模式

- 调试：在应用程序中查找并修改错误的过程称之为调试。
- 1. 错误类型
- Visual Basic应用程序中出现的错误分为三类：编译错误、运行时错误和逻辑错误。

9.1 错误类型和程序模式

- (1) 编译错误
- 是指在程序编译过程中出现的错误。
- 产生原因：通常是由于不正确书写代码而产生的。例如关键字写错、遗漏标点符号等。
- Visual Basic利用自动语法检查功能找出有语法错误的语句，帮助用户纠正语法错误。
- 自动语法检查是在“工具/选项/编辑器”中设置，设置好自动语法检查功能后，只要在代码窗口中出现一个有语法错误的语句，当将光标移到其它语句行时就会立即显示错误提示信息。如图所示。

9.1 错误类型和程序模式



The image shows a screenshot of the Microsoft Visual Basic development environment. The main window is titled "工程1 - Form1 (Code)". The code editor displays the following code:

```
Private Sub Form_Load()  
Dimm i As Integer  
  
End Sub
```

The word "Dimm" is highlighted in red, indicating a syntax error. A dialog box titled "Microsoft Visual Basic" is overlaid on the code editor, displaying a yellow warning triangle with a black exclamation mark. The text in the dialog box reads:

编译错误:
缺少: 语句结束

At the bottom of the dialog box, there are two buttons: "确定" (OK) and "帮助(H)" (Help).

9.1 错误类型和程序模式

- (2) 运行时的错误

- 是指应用程序在运行期间执行了非法操作所产生的错误，例如除法运算中除数为零，打开文件时文件没找到，访问磁盘时驱动器中无盘等。

- 例9.1：下标越界错误

```
Private Sub Command1_Click()  
    Dim A(10) as Integer  
    Dim I as Integer
```

9.1 错误类型和程序模式

```
For I=0 to 20
```

```
  A(I)=I
```

```
Next
```

```
End Sub
```

如图所示

9.1 错误类型和程序模式

Microsoft Visual Basic

实时错误 '9':

下标越界

继续 (C)

结束 (E)

调试 (D)

帮助 (H)

9.1 错误类型和程序模式

- (3) 逻辑错误
- 当应用程序未按预期方式执行时，就会产生逻辑错误。
- 例9.2 某程序如下：

```
Private Sub Command1_Click()  
    Dim sumnumber As Integer  
    Dim I As Integer  
    For I=0 to 100  
        sumnumber=sumnumber+I  
    Next  
    Print sumnumber  
End sub (变量名写错)
```

9.1 错误类型和程序模式

- 2. 应用程序的模式
- 有三种模式：设计模式，运行模式和中断模式。
- (1) 设计模式
- 启动Visual Basic6.0后就进入设计模式。
- 功能：设计窗体布局、绘制控件、编写代码、设计并查看属性等。另外还可以在代码窗口中设置断点，创建监视表达式，但不能在设计模式下使用调试工具。

9.1 错误类型和程序模式

● (2) 运行模式

- 单击工具栏中的启动按钮或者选择“运行”菜单中的“启动”选项，都可以进入运行模式。

- 功能：同应用程序交换信息，但不能修改程序。

● (3) 中断模式

- 功能：分析应用程序的当前状态并修改程序代码，并且所有的调试工具都是在中断模式下运行的。

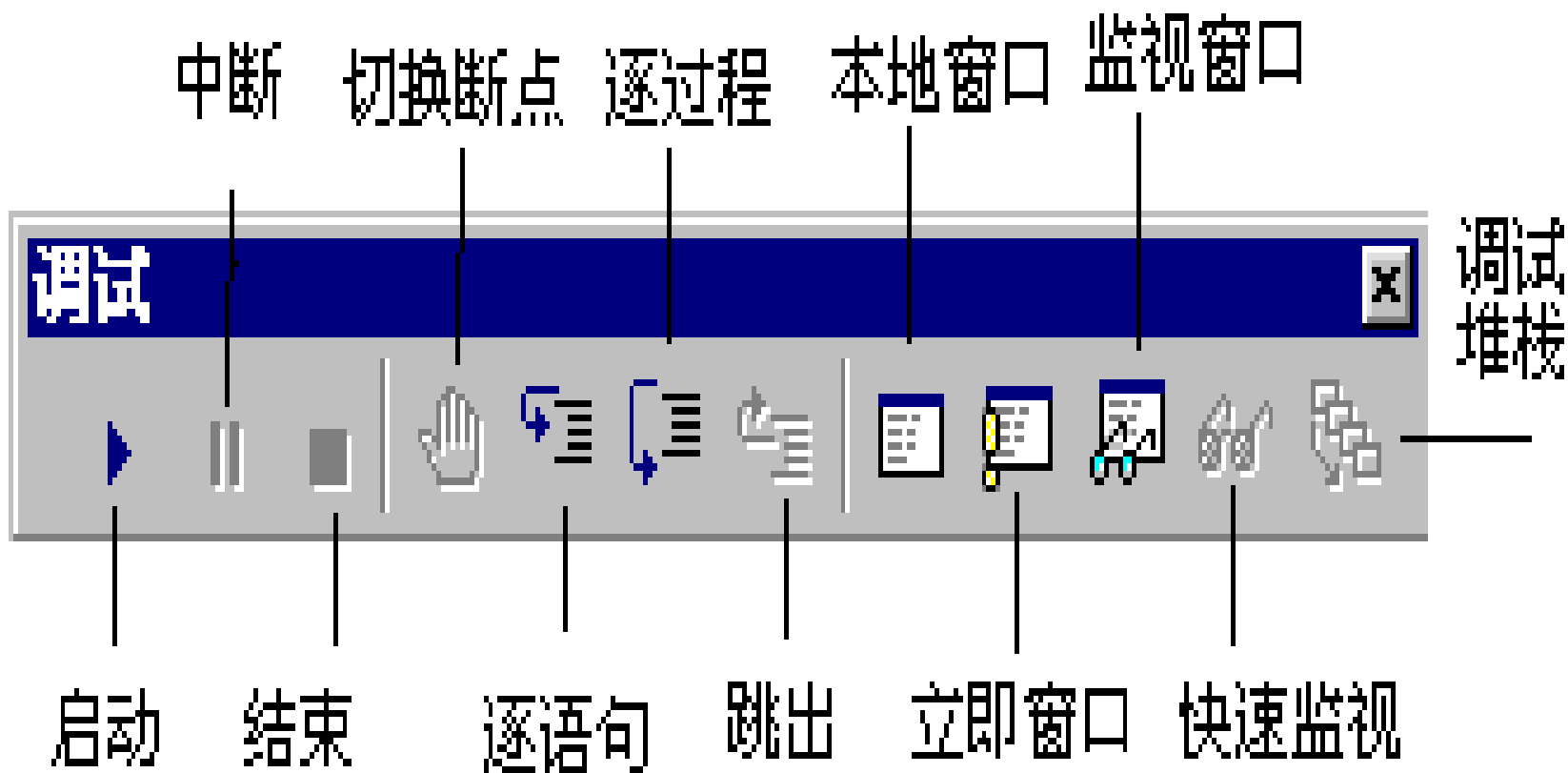
9.1 错误类型和程序模式

- 进入中断模式有几种方式：
- 在设计模式下通过设置断点或者使用“stop”语句将应用程序置于中断模式
- 选择“运行”菜单中的“中断”选项，或者单击“中断”按钮将应用程序置于中断模式
- 按下Ctrl+Break键来引导程序由运行模式切换到中断模式。
- 应用程序在运行时产生错误，也可以自动切换到中断模式。

9.2 程序调试

- 是指在应用程序中查找并修改错误的过程。
- 1. 调试工具
- 调试工具有：切换断点、逐语句、逐过程、跳出，本地窗口、立即窗口、监视窗口、快速监视和调试堆栈。
- 通过“调试”菜单或调试工具栏可以获得全部调试工具。如图所示。

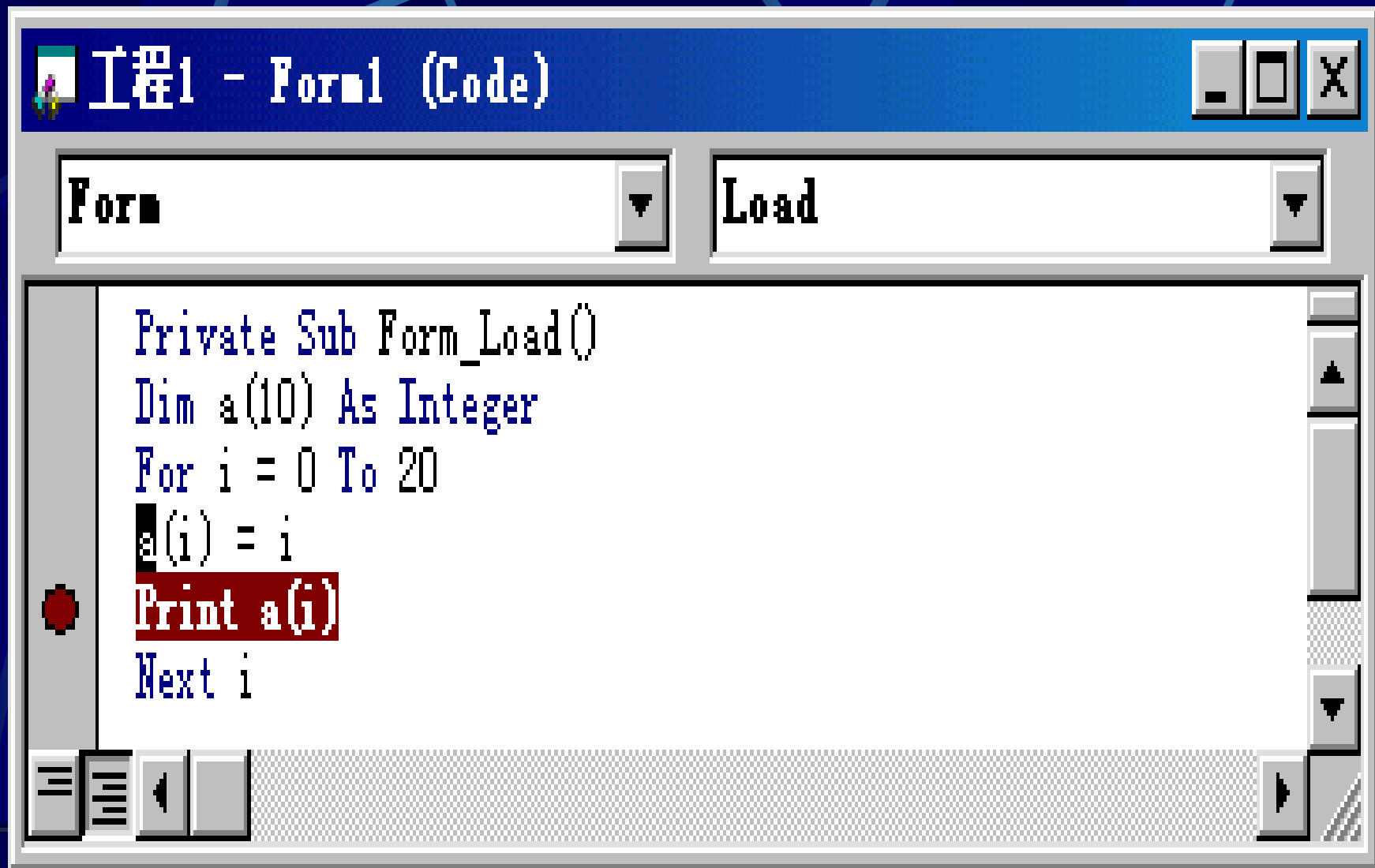
9.2 程序调试



9.2 程序调试

- (1) 切换断点
- 运行一个程序时，遇到具有断点的代码会中断应用程序的执行。
- 断点设置位置：通常断点被设置在代码中被怀疑可能会出问题的区域，断点可以在中断模式或设计模式下设置。
- 设置断点的方法：通常是在代码窗口中，在要设置断点的那一行代码的左边空白区单击鼠标左键设置。如图所示。

9.2 程序调试



9.2 程序调试

- 设置断点后，Visual Basic将以粗体突出显示设定行，并在该行前面显示一个黑色（或红色）圆点符号。
- (2) 逐语句：逐语句执行就是一条语句一条语句的执行代码，通过查看应用程序的窗体或调试窗口来判断这条语句是否正确。如果执行的代码是过程调用，也会跟踪到被调用过程中继续一条语句一条语句的执行。
- (3) 逐过程：逐过程执行就是以整个函数或过程为一个整体，一次执行下去。

9.2 程序调试

- 逐过程与逐语句的相似之处：也是一条语句一条语句的执行代码。
- 不同之处是：当执行的代码是过程调用时，逐语句会跟踪到被调用过程中，而逐过程不会跟踪到被调用过程中，它把被调用过程作为一个整体一次执行过去。

9.2 程序调试

- (4) 跳出：当用逐语句方法跟踪进入过程或函数中，如果发现过程中的语句没有问题，可以单击“调试”工具栏的“跳出”按钮，从当前的过程中跳出，去执行过程调用者的下一条语句。
- (5) 本地窗口：用户在调试程序时可以利用“本地”窗口显示当前过程中所有变量的值。当程序的执行从一个过程切换到另一个过程时，“本地”窗口的内容会发生改变，它显示的是当前过程中可用的变量。如图所示。

9.2 程序调试

The screenshot shows a 'Locals' window in a debugger. The title bar reads '本地' (Local). The main area shows the current object '工程1.Form1.aa'. Below this is a table with three columns: '表达式' (Expression), '值' (Value), and '类型' (Type). The table lists the following variables:

表达式	值	类型
<input checked="" type="checkbox"/> Me		Form1/Form1
x	0	Integer
aa	0	Integer
aa	0	Integer
k	0	Integer

9.2 程序调试

● (6) 立即窗口

● 功能：可以检查某个属性或者变量的值，还可以执行单个的过程，对表达式求值，或为变量或属性赋值等。

● 显示立即窗口的方法：单击“调试”工具栏上的“立即窗口”按钮。如图所示。

。

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：<https://d.book118.com/326112034020010241>