

第 1 章 Visual Basic 6.0 概述

[课程引入]:

先用教师机播放《VB教学实例程序演示系统》软件，将几个用VB编写的、有趣的典型应用程序实例，例如：倒计时、滚动字幕、学生学籍管理、自制IE浏览器等演示给学生，引入本课程，以提高学生的感性认识及学习兴趣。

[课堂提问及讨论]:

有哪些同学见过或使用过VB应用程序？试举例说明。

[知识点讲解]:

§ 1.1 程序设计基础

一、程序设计几个概念

1、程序：是为解决某一问题而编写的语句集合，是将一个实际问题的具体操作用某种程序设计语言描述出来的指令序列。

2、程序设计：是分析问题、编写程序、调试程序的过程。

3、程序设计语言：是人们根据描述问题的需要而设计的，有固定的符号和语法规则、是计算机能够识别的人机交流所使用的语言。

二、程序设计语言的发展概况

1、机器语言

2、汇编语言

3、高级语言

§ 1.2 Visual Basic 6.0 简介

一、Visual Basic 6.0 简介

1、概念：Visual Basic 6.0 简称 VB，是 Windows 环境下可视化的、面向对象的程序设计开发工具。

其中：Visual 英文原意是“可视的”、“视觉的”。6.0 是版本号。

2、VB 功能：用它可以开发数学计算、数据库管理、客户/服务器软件、通讯软件、多媒体及 Internet/Intranet 软件等。其功能之强大绝非是早期 Basic 所能比拟的，其直观、通俗易懂，特别适合于初学者。

3、VB 发展过程：美国微软公司于 1991 年推出 VB1.0

版至今已经历了 6 个版本。

4、VB 中常用的几个基本概念：

面向对象：面向对象编程是建立在类的基础上的，其通过类的封装而使源程序更加便于维护。

事件驱动：在事件驱动模式下，程序的执行是依靠系统能够被识别的触发事件启动的。

5、VB 的版本：

VB6.0 是 Microsoft Visual Studio 开发工具组件中的一员，它共有 3 个版本：标准版、专业版、企业版。教学中采用企业版。

6、运行环境

操作系统：NT Server 4.0、Windows 98，或更高版本。

CPU：486DX/66 MHz 或更高的处理器。

内存：Windows 95 / 98 下要求 16MB 以上的内存，Windows NT 下要求 32MB 以上。

磁盘空间：对于标准版或专业版，典型安装 48MB，完全安装 80MB。对于企业版，典型安装 128MB，完全安装 147MB。

其他设备：CD-ROM 驱动器，鼠标。

7、VB 特点：

- ◇具有面向对象的可视化设计工具；
- ◇事件驱动的程序运行机制；
- ◇提供了易学易用的应用程序集成开发环境。
- ◇结构化的程序设计语言；
- ◇支持多种数据库系统的访问；
- ◇支持动态数据交换、动态链接库和对象嵌入技术；
- ◇完备的联机帮助功能。

8、VB 安装（略）、帮助系统的使用（略）

二、Visual Basic 6.0 启动、退出

1、VB 的启动

双击桌面上的 VB 快捷方式图标，或单击桌面上“开始”按钮→“程序”→Microsoft Visual Basic 6.0 中文版→Microsoft Visual Basic 6.0 中文版，都可以启动 VB6.0。

2、VB 的退出

单击关闭按钮、ALT+F4 快捷键、或通过菜单命令“文件\退出”均可关闭并退出 VB 集成环境。

[课堂练习]：

1、课后习题 1：2。

2、预习本章后面内容，思考课后习题 1：3、6。

[答疑]：针对学生提出的问题答疑。

[小结]：本节主要讲解 Visual Basic 6.0 可视化编程的基本概念、发展过程、特点及其启动、退出的方法。要求掌握全部知识点。

[作业]：上机熟悉 Visual Basic 6.0 的启动、退出及界面组成。

预习第 2 章内容。

§ 1. 3 VB 的可视化开发环境

[复习]: 以提问的方式, 复习上节所讲知识, 检查学生掌握情况。

[引入]: 启动 VB 后, 同学们见到的 VB 开发环境由哪几部分组成? 这个问题即是本节要讲的内容。

[知识点讲解]:

一、VB 的界面组成

VB 的界面由标题栏、菜单栏、工具栏、控件箱、窗体编辑窗口、工程资源管理窗口、属性窗口、窗体布局窗口等组成。(如教材图 2-2 所示)

二、VB 各部分功能简介

1、菜单栏: 是在集成开发环境下发布命令的最基本手段。Visual Basic 顶级菜单有: 文件、编辑、视图、工程、格式、运行、查询、图表、工具、外接程序、窗口、帮助, 但菜单命令是通过子菜单中的子菜单项发出的。

文件菜单: 主要用于建立、打开、添加、移去、保存工程和文件, 包括: 新建工程、打开工程、添加工程、移除工程、保存工程、工程另存为、保存文件、文件另存为、打印、打印设置等子菜单项。

编辑菜单: 在对工程进行修改时, 编辑菜单用于各种编辑操作。包括: 撤消、重复、剪切、复制、粘贴、粘贴链接、删除、全选、查找、缩进、凸出、插入文件、属性方法列表、快速信息、参数信息书签等子菜单项。

视图菜单: 视图菜单用于显示各种窗口及和窗口有关的操作。包括: 代码窗口、对象窗口、定义、最后位置、对象浏览器、立即窗口、本地窗口、监视窗口、调用堆栈、工程资源管理器、属性窗口、窗体布局窗口、属性页、表、缩放、显示窗格、工具箱、调色板、工具栏等子菜单项。

工程菜单: 用于为当前工程创建模块、作对象引用或提供各种设计器。包括: 添加窗体、添加 MDI 窗体、添加模块、添加用户控件、添加属性页、添加用户文档、添加设计器、添加文件、移除、引用、部件、工程属性等子菜单项。

格式菜单: 用于界面设计, 能使界面中的控件按规范排列。包括: 对齐、统一尺寸、按网格调整大小、水平间距、垂直间距、锁定控件等子菜单项。

调试菜单：用于调试、监视程序。包括：逐语句、逐过程、跳出、运行到光标处、添加监视、编辑监视、快速监视、切换断点、清除所有断点等子菜单项。

查询菜单：各条命令用于数据库表的查询及相关操作。所提供的各种查询设计工具，使用户能够通过可视化工具创建 SQL 语句，实现对数据库的查询、修改。

运行菜单：用于执行程序。包括：启动、全编译执行、中断、结束、重新启动等子菜单项。

工具菜单包括：添加过程、过程属性、菜单编辑器、选项、发布等子菜单项。

外接程序菜单用于为当前工程创建含有：可视化数据管理器、外接程序管理器等子菜单项。

窗口菜单：用于调整已打开窗口的排列方式。包括：拆分、水平平铺、垂直平铺、层叠、排列图标等子菜单项。

帮助菜单：用于给用户提供各种方式的帮助。包括：内容、索引、搜索、技术支持等子菜单项。

[课堂练习 1]：

提问：若将界面上的某窗口误操作关闭，如何通过菜单将其重新打开？

2、Visual Basic 6.0 工具栏

工具栏的作用是可以单击其上面的图标按钮执行菜单命令，由此加快操作速度。

Visual Basic 能提供“编辑”、“标准”、“窗体编辑器”、“调试”等几个工具栏，用户也可以按自己的需要，通过“视图\工具栏”菜单来选取“自定义”工具栏。

3、Visual Basic 6.0 控件箱（又称工具箱）

控件箱是 Visual Basic 为程序界面开发而提供控件的面板。默认为内置控件。

除了 Visual Basic 内置控件之外，用户还可以通过菜单“工程\部件”打开“部件”对话框，从中能添加控件、设计器或可插入对象到工具箱中，也可以引用已加载的控件工程。

4、工程管理器窗口

工程管理器又称工程资源管理器，它以层次列表的方式管理当前工程文件或工程组文件中所包含的窗体文件、模块文件、类模块文件等 3 种类型的文件。

5、属性窗口

属性窗口用于显示和设置窗体或控件的所有属性。它由 3 部分组成：对象列表框、属性列表框、属性含义说明。

6、窗体布局窗口

窗体布局窗口用于确定程序运行时窗体在屏幕上的初始位置。

除了以上基本部分外，在 VB 程序设计中还有一个重要的窗口，即代码编辑窗口。

[课堂练习 2]:

试设置默认窗体的背景颜色为红色，字体的颜色为黄色，试着在它上面“画”几个控件。

[答疑]: 针对学生提出的问题答疑。

[小结] 本章节主要讲 Visual Basic 6.0 集成开发环境界面组成。本节内容可作为“字典”备查，不需全部掌握，编程用到时，再返回本节查看。重点掌握：文件、运行菜单；控件箱、属性窗口的使用即可。

[作业] 上机熟悉 Visual Basic 6.0 的启动、退出及界面组成，控件箱、属性窗口的使用。

§ 1.4 VB 可视化编程概念

§ 1.5 可视化编程初识

[复习]：上节所讲 Visual Basic 6.0 集成开发环境界面组成。

[实例引入]：VB 的第一个实例——“欢迎进入奇妙的 VB 世界”。先演示此例的运行结果，用以引入本节内容。

[知识点讲解]：

一、VB 编程中用到的几个概念

1、对象：(object)是代码和数据的集合，在 VB 环境中，常用的对象有工具箱中的控件、窗体、菜单、应用程序的部件，以及数据库等。

对象有三个特征：属性、事件和方法。

对象的属性：VB 中每个对象都有一组特征，这组特征称为属性。

属性设置方法：先选定对象，然后在属性窗口中找到相应属性直接设置；或编程在代码中设置，格式为：对象名·属性名=属性值。

2、事件：对象的事件(Event)就是对象上所发生的事情。VB 中事件是预先定义好的，能够被对象是别的动作，如：单击事件(Click)、双击事件(Dblclick)、装载事件(Load)、鼠标移动事件(MouseMove)等。对象的事件是固定的，用户不能建立新的事件，程序员只需编写必须相应的事件过程即可。

事件过程的一般编写格式为：

Sub

.....

程序代码

.....

End sub

3、方法：事件的方法与事件过程类似，它可能函数或过程，用于完成某种特定功能，而不能相应某个事件。如：对象打印方法(Print)、窗体显示方法(Show)、移动方法(Move)等。可用以下格式调用对象方法：对象名·方法名。

例如：Form1·Show

二、VB 可视化编程的步骤

一般步骤为：打开 VB→新建工程→添加控件、设计应用程序界面→设置属性→编写代码→运行工程→修改工程→保存工程。

1、新建工程：

步骤 启动 VB→新建工程,标准 EXE。或者文件菜单→新建工程→标准 EXE。

2、添加控件、设计应用程序界面：

方法：单击工具箱中的某控件图标，鼠标变成“+”指针，然后在窗体工作区按住鼠标左键拖动鼠标，松开即可在窗体上画出对应控件。

3、属性设置：先选定对象，然后在属性窗口中找到相应属性直接设置；或编程在代码中设置，格式为：对象名·属性名=属性值。

4、编写代码：打开代码编辑器，单击“对象”下拉列表框，从中选取相应对象，单击“过程”下拉列表框，从中选取相应事件，然后编写代码。

5、运行工程：单击工具栏上的“运行”按钮或按 F5 键，即可运行工程。单击“退出”按钮或标题栏上的“关闭”按钮，即可结束工程运行。

6、保存工程：单击“文件”菜单→“保存工程”或“工程另存为”或单击工具栏上的“保存工程”，即可保存工程。

三、实例：

实例 1：VB 的第一个实例——“欢迎进入奇妙的 VB 世界”。运行结果如图 2-8 所示。程序界面由一个文本框和一个命令按钮组成，单击命令按钮时，文本框会出现“欢迎进入奇妙的 VB 世界”字样。详见教材 P10-12。

具体操作步骤如下：

1、设计界面：

在窗体上画一个一个文本框和一个命令按钮。

2、设置属性：

按表 2-1 设置各对象的属性：

控件名	属性名	属性设置值
窗体 Form1	caption	欢迎进入奇妙的 VB 世界

文本框 Text1	text	
命令按钮 Command1	caption	确定

3、编写代码：

命令按钮 Command1 的 click 事件过程代码为：

```
text1.text=" 欢迎进入奇妙的 VB 世界"
```

4、运行应用程序：

选择“运行|启动”、F5 键、单击工具栏上“启动”按钮都可以运行该示例程序。

5、保存工程：

单击工具栏上的“保存”按钮即可保存该工程和窗体文件。

(详细讲解并演示以上几个步骤的制作过程)

[课堂练习]：上机实现以上实例。

[答疑]：针对学生提出的问题答疑。

[小结] 可视化编程的概念、程序调试步骤。

习题 1：

思考：习题 2、3、5、6、7；

上机：熟悉 VB 的界面，编程并上机实现：习题 7。

§ 1.5 可视化编程初识

[复习及作业讲解]：

- 1、复习 VB 应用程序调试的全过程。
- 2、讲解作业 7 题，并以此引入“工程”的概念。

[知识点讲解]：

一、VB 的开发模式

VB 提供了 3 种开发模式：设计模式、运行模式和中断模式。

二、VB 工程概述

1、工程文件(.vbp)：VB 使用了称为“工程”的特殊文件来管理应用程序中所有不同的文件。它提供了与该工程有关的全部文件和对象的清单，每当保存工程时，VB 都要更新工程文件。一个完整的工程中的文件包括：

- ◇跟踪所有部件的工程文件 (.vbp)；
- ◇窗体文件 (.frm)；
- ◇窗体所对应二进制数据文件 (.frx)；
- ◇标准模块文件 (.bas)。

2、工程管理

- (1) 新建工程：“文件|新建工程”、选择类型后“确定”。
- (2) 打开工程：“文件|打开工程”。
- (3) 保存工程：两种方法：直接保存和工程另存为。
- (4) 向工程中添加文件：“工程|添加文件”将现存的文件添加到当前的工程文件中。
- (5) 从工程中删除文件：在工程资源管理器中先选定要删除的文件，然后再选择“工程|移除文件”移除。

3、启动窗体的设置：

可以将工程中的任一个窗体设置为启动窗体，方法是：“工程|xxx 属性…”，其中 xxx 为当前工程的名称。

[课堂练习]：查看上节实例（第 1 个 VB 实例）所生成的工程文件及其他文件，注意其扩展名。

[答疑]：针对学生提出的问题答疑。

[小结]： 本节主要讲解工程的概念、工程文件的组成、工程的管理等。要求了解工程文件的组成，掌握工程文件的保存、打开方法。

[作业]：P18，习题 2。

笔答：习题 1、2、3、4、6、10；

上机：熟悉 VB 的工程，编程并上机实现：习题 13。

[拓展与提高]：试将习题 13 贴上一幅自己喜欢的图片。

第 3 章 VB 简单应用程序的设计

[复习、提问]：以提问方式复习上节所讲 Visual Basic 6.0 工程知识。

[实例演示、引入]：前面章节实例中用到的命令按钮、文本框等统称为——控件。在窗体上任意画若干个控件，演示其常见属性的设置，大小、位置的调整。用以引入本节内容。

[知识点讲解]：

§ 3.1 控件概述

一、控件概述

1、控件：是建立 VB 应用程序的基本元素。可视化程序设计就是通过控件实现的。

2、VB 的控件分类：分为三大类：内部固有控件、ActiveX 控件、可插入的对象。用户可以通过“工具箱”获取各种内部控件。

内部控件是包含在 Visual Basic 系统内，可以直接使用的控件；因此具有相对较好的运行性能。

3、常用的内部控件有：命令按钮、文本框、标签、框架、列表框、组合框、复选框、选项按钮、图片框、定时器、数据控件等。

§ 3.2 控件的常用属性

1、属性的设置方法：

有两种方法：一种是通过属性窗口来直接设置，另一种是在代码中设置。即：
对象名称. 属性名称=设置值

例如：Command1. Caption=“确定”

用代码设置的方法常用于动态地修改控件的有关属性。

2、控件的常用属性

Caption属性：也称标题属性，用于设置在控件上显示的文本内容。

Name属性：对象在创建时所取的名字。编代码引用该对象时使用，这个名字不会出现在窗体上，且每个对象都有。

Enabled属性：决定了对象在运行时是否允许用户操作。

Font属性：用于设置字符的格式，如字体、字号、字形等。

Height、Width、Top、Left 属性：决定控件的大小、位置。

ForeColor、BackColor：分别设置对象的背景色、前景色。

§ 3.3 控件的格式化

1、控件的添加/删除：

两种方法：单击拖动、双击再移动。

2、控件的大小与位置：

控件的位置调整有两种方法：一是用鼠标拖动对象，二是使用 Ctrl+方向键。

3、窗体与控件的布局调整：

两种方法：一是通过“格式”菜单，二是通过“窗体编辑器”。

[课堂练习]：上机练习：在窗体上任意画若干个控件，试完成其常见属性的设置，大小、位置的调整。

§ 3.4 基本控件的使用

一、标签 (Label)

1、Label (标签) 控件是专门用来显示文本的，因此用户不能对标签内的文本进行编辑。

2、一般被用作标题名。通常用标签来标注本身不具有 Caption 属性的控件。例如，可用 Label 控件为文本框、列表框、组合框等控件添加描述性的文字。

3、标签常用的属性有：

Caption 属性：缺省属性，设置标签中显示的文本内容。

AutoSize 属性：设置控件是否能自动调整以显示所有的内容。

WordWrap 属性：该属性设置是否扩大以显示标题文字。

二、命令按钮 (CommandButton)

1、当用户需要交互式地控制应用程序的时候，最简便的方法就是给用户提供一个“命令按钮”。

2、命令按钮的作用是：在用户单击它时，将会激发它的 Click 事件。因此，将代码写入 命令按钮的 Click 事件过程，通过用户单击就可以执行相关操作。

3、命令按钮的常用属性有：

Cancel 属性：用于设置按钮为取消按钮。

Caption 属性：用于设置在命令按钮上显示的文本。

Style 属性：设置命令按钮的外观。

三、文本框（**TextBox**）

1、**TextBox**（文本框）控件一般是用来接收或显示输入输出信息，可用于编辑文本。

2、经常使用的文本框属性有：

Text 属性：用于设置显示的文本内容。

MaxLength 属性：用于设置文本框中输入字符串的长度限制。

MultiLine 属性：用于设置文本框是否以多行方式显示文本。

PasswordChar 属性：用于设置是否显示用户键入的字符。

ScrollBars 属性：用于设置文本框是否有垂直或水平滚动条。

[答疑]：针对学生提出的问题答疑。

[小结]本节主要讲三种基本控件的编程使用。要点：三种基本控件的常用属性的设置和编程实现。

[习题]

预习：实例 1、2、3。

思考：习题 3：2、4、5。

§ 3.4 基本控件的使用——例题

[提问]：以提问方式复习上节所讲 VB 6.0 控件知识。

[实例演示]：演示实例 3-1，用以引入本节内容。

[知识点讲解]：

例题讲解：

1、讲解并演示教材 P27 例 3-1。

实现步骤、编程代码：如教材 P27（略）

[课堂练习 1]：上机实现例 3-1。

2、讲解并演示教材 P28 例 3-2。

[课堂练习 2]：上机实现例 3-2。

3、演示教材 P31 例 3-3。

[课堂练习 3]：上机实现例 3-3。

[答疑]：针对学生提出的问题答疑。

[小结]本节主要以实例形式讲解三种基本控件的编程思路及编程方法。要求掌握重点语句的使用。

[习题]习题 3 2、4、5。

编程并上机实现：7、11、12。

要求：交上机作业。每人通过网络上传一个文件夹，文件夹以班级、姓名、学号为名字，内含 3 个可执行文件（EXE 文件）。记一次平时成绩。

“基本控件的使用”——习题课

[学生演示]：指定学生演示实例 3-1、3-3，检查学生例题掌握情况，并给其记一次平时成绩。

[学生作业讲评]：针对所交作业中出现的共同问题讲评。

[知识点讲解]：

例 1：课后习题 P33-7。

设计步骤（同以前例题，略），程序代码如下：

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, X
As Single, Y As Single)
    If Button = 1 Then
Form1.Picture=LoadPicture("c:\programfiles\microsoftoffice\office10\
bitmaps\styles\globe.wmf")
    Else
        Form1.Picture = LoadPicture("")
    End If
End Sub
```

[课堂练习 1]：学生上机实现例 1。

例 2：程序结果为分别在窗体上显示“今天是：”后分别显示当前年月日、时分秒。制作步骤（略），程序代码如下：

```
Private Sub Command1_Click()
Label1.Caption = "欢迎使用：" & Chr(13) & Chr(13) & "VB 6"
End Sub

Private Sub Command2_Click()
Label1.Caption = "今天是：" & Chr(13) & Chr(13) & Year(Date) & "年"
& Month(Date) & "月" & Day(Date) & "日" & Chr(13) & Chr(13) & Hour(Time)
```

```
& "时 " & Minute(Time) & "分" & Second(Time) & "秒"  
End Sub
```

```
Private Sub Command3_Click()  
Unload Me  
End Sub
```

[课堂练习 2]：学生上机实现例 2。

例 3： P33 习题 12，程序结果如图 3-10，程序代码如下

```
Private Sub Command1_Click() ‘红灯  
Label1.Picture=LoadPicture("c:\programfiles\Microsoft          visual  
studio\common\trff10c.ico")  
Label2.Visible=false  
Label2.Visible=false  
End Sub
```

另两段代码类似（略）。

[课堂练习 3]：学生上机实现例 3。

[答疑]：针对学生提出的问题答疑。

[小结]本节主要通过课后习题讲解，使用学生训练掌握三种基本控件的编程知识及技巧。

[习题]课后实验 1，P243 实验二、2(1)-(2)。

要求：交上机作业 P243 实验二、2(1)-(2)。每人通过网络上传一个文件夹，文件夹以班级姓名学号为名，内含 2 个可执行文件（EXE 文件）。

[学生演示]: 指定学生演示课后实验 1 , P243 二、2 (1) - (2)。, 检查学生作业情况, 并给其记一次平时成绩。

[学生作业讲评]: 针对以上演示中出现的问题讲评。

[知识点讲解]:

第 4 章 代码基础及程序结构

§ 4.1 数据类型

一、常量与变量

1、常量: 是用标识符表示永远不变的数值或字符串。

值得注意的是: 常量不能象对变量那样在程序运行过程中给常量标识符赋以新值。

常量有两种来源: 内部的或系统定义的, 其由 Visual Basic 系统或引用的对象提供; 用户自定义的, 其需要用 Const 语句来声明。

用户自定义常量:

用户自定义常量在使用前需要声明, 由此使该标识符能够被程序识别。其语法是:

[Public/ Private] Const 常量名[As 类型] = 常量表达式

例如:

```
Const conPi = 3.14159265358979
```

```
Const conMaxPlanets As Integer = 9
```

一旦已定义常量, 就可将其放置在代码中 使用。

2、变量:

在 Visual Basic 中执行应用程序期间, 用变量临时存储数据。变量有名字(变量标识符)和数据类型(确定变量能够存储的数据的种类)。

可以把变量看作内存中存放未知值的所在处。

声明变量 : 声明变量就是事先将变量通知程序, 由此使变量的使用合法。

语法: Dim/Private/Public/ Static 变量名 [As 类型]

例如:

```
Dim I As Integer
```

```
Private YourName As String, BillsPaid As Currency
```

```
Public Test, Amount, J As Integer
```

变量范围：变量的范围确定了能够知晓该变量存在的那部分代码。

一个变量在划定范围时被看作是过程级变量、模块级变量，或是公用变量。这取决于声明该变量时采用的方式。

过程级变量：过程级变量只有在声明它们的过程中才能被识别，也称它们为局部变量。局部变量只能用 `Dim` 或 `Static` 关键字来声明它们。例如：

```
Dim intTemp As Integer
```

```
Static intPermanent As Integer
```

用 `Dim` 声明的变量只在过程执行期间才存在。

而用 `Static` 声明的局部变量，则在整个应用程序运行期一直存在，称做：静态变量。

模块级变量：模块级变量对该模块（包括：标准模块、窗体模块、类模块）的所有过程都可用，但对其他模块的代码不可用。可在模块顶部的声明段用 `Private` 或 `Dim` 关键字声明模块级变量。例如：

```
Private intTemp As Integer
```

公用变量：为了使模块级的变量在其他模块中也有效，需要用 `Public` 关键字声明。经过 `Public` 关键字声明的变量是公用变量，其值可用于应用程序的所有过程。只能在模块的声明段中用 `Public` 关键字声明公用变量，例如：

```
Public intTemp As Integer
```

二、数据类型：

所有变量都具有数据类型，以决定能够存储哪种数据。变量的数据类型决定了如何将代表这些值的位存储到计算机的内存中。

除变量外，数据类型也用于其他场合。如在给对象属性赋值时，这个值就有数据类型；函数的参数也有数据类型。事实上，在 `Visual Basic` 中，凡是涉及数据就与数据类型有关，包括：变量、常量、数组，以及函数、过程中的参数。

1、数值型数据

Visual Basic 支持几种数值数据类型— Integer（整型）、Long（长整型）、Single（单精度浮点型）、Double（双精度浮点型）、Currency（货币型），另外，Byte（字节型）也可用于数值计算。

所有数值变量都可相互赋值。在将浮点数赋予整数之前，Visual Basic 将浮点数的小数部分四舍五入到整数。

2、String 数据类型

String 型变量用于存储字符串，如下列语句：

```
Dim S As String
S = "Database"
```

在缺省情况下，String 型变量是一个可变长度的字符串变量，随着对其赋予新数据，它的长度可增可减。但也可以声明具有固定长度 String 型变量，语法是：

```
Dim/Private/Public/ Static variablename String * size
```

3、Boolean 数据类型

若变量的值只是“true/false”、“yes/no”、“on/off”信息，则可将它声明为 Boolean 类型。Boolean 的缺省值为 False。

4、Date 数据类型

用于存储 Date 和 Time 值，Visual Basic 接受许多种文字的 Date 和 Time 格式。例如：

```
Dim SomeDate As Date
SomeDate = #1980-03-16#
SomeDate = #3-6-93 13:20#
SomeDate = #March 27 , 1993 1:20am#
```

5、Object 数据类型

Object 变量采用 32 位（4 个字节）地址来存储对象，该地址可引用应用程序中或某些其他应用程序中的对象。可以随后（用 Set 语句）指定一个被声明为 Object 的变量去引用应用程序所识别的任何实际对象，如：

```
Dim objDb As Object
```

```
Set objDb = OpenDatabase ("c:\Vb5\Biblio.mdb")
```

6、Variant 数据类型

如果把数据赋予 Variant 变量，则不必在这些数据的类型间进行转换。Visual Basic 会自动完成任何必要的转换。例如：

```
Dim SomeValue
```

```
SomeValue = "17"
```

```
SomeValue = SomeValue - 15
```

```
SomeValue = "U" & SomeValue
```

7、自定义数据类型

在模块的声明部分可以使用 Type 语句建立自定义数据类型。

语法格式：

```
[Private|Public] Type 自定义类型名
```

```
元素 As 类型
```

```
...
```

```
End Type
```

[小结]：本节主要讲解常量、变量，几种数据类型的概念。要求学生理解。

[作业]：思考：理解几种数据类型。

[复习]: 几种数据类型。

[知识点讲解]:

§ 4.2 运算符和表达式

一、运算符

程序中对数据的基本加工, 是依靠运算符实现的。在 Visual Basic 中运算符的类型有算术运算符、字符串运算符、关系运算符、逻辑运算符。

1、算术运算符

用来进行算术运算, 有: + (加)、- (减)、* (乘)、/ (除)、\ (整除)、Mod (求模)、^ (乘方) 等运算符。

2、字符串运算符

&、+ 运算符都能用于进行两个字符串的连接, 但 & 运算符强制两个表达式按字符串连接, 尽管被连接的表达式不是字符串也是如此。而 + 运算符则较复杂, 当两个表达式都是字符串时按字符串连接, 否则视具体情况会有不同结果。因此, 在进行字符串连接时一般使用 & 运算符。

3、运算符

关系运算符包括: < (小于)、<= (小于或等于)、> (大于)、>= (大于或等于)、<> (不等于) 和 = (等于), 用来反映两个数值或字符串表达式之间的关系。关系成立, 返回 True (真); 关系不成立, 返回 False (假); 两个表达式中若有 Null, 则返回 Null。

4、逻辑运算符

逻辑运算符包括: Not (非)、And (与)、Or (或)、Xor (异或)、Equ (逻辑等于) 和 Imp (逻辑蕴涵), 用于表达两个逻辑表达式之间的关系。在进行逻辑运算时, 只要参入运算的表达式中有一个为 Null, 则将返回 Null。

5、运算符优先顺序

在表达式中, 当运算符不止一种时, 要先处理算术运算符, 接着处理比较运算符, 然后再处理逻辑运算符。

算术运算符优先级由高至低是: 指数运算 (^)、负数 (-)、乘法和除法 (*、/)、整数除法 (\)、求模运算 (Mod)、加法和减法 (+、-)、字符串连接 (&)。

逻辑运算符优先级由高至低是：Not、And、Or、Xor、Eqv、Imp 。

所有比较运算符的优先顺序都相同 。

二、表达式

通过常量、变量、运算符、函数等的组合，由此构成表达式。

Visual Basic 中的赋值运算符“=”用来给变量、数组或对象的属性赋值，即把运算符右边的内容（包括常量、变量、函数返回值或对象的属性）赋给运算符左边的变量或属性。

语法格式： Variable = 表达式

其中 variable 可以是变量、数组的元素等。表达式可以是常数、常量、变量或函数调用，但必须有确定的值。

[例题]计算下列表达式的值。

解： $2+3>5$ and $5<3$ 值为 false

not $5<3$ and $6-2=10+2$ 值为 true

[答疑]：针对学生提出的问题答疑。

[小结]：本节主要讲解几种运算符的作用及优先级。要求掌握并能够正确计算各类表达式的值。

[作业]：P62 习题 5、6、7、8、9、10、11、12。

§ 4.3 程序流程

[复习]: 几种运算符的作用及优先级。

[讲解习题]: 讲解上节课后习题。

[实例引入]: 演示例题: P66-3, 以引入程序流程概念。

[知识点讲解]:

一、顺序流程控制

程序的基本运行方式是自顶向下地顺序执行各条语句, 被称做顺序结构。对于简单的问题, 依靠顺序结构就可以完成任务; 但对于稍微复杂一点的问题, 单纯依靠顺序结构也许就不够了, 而必须在程序中加入更多的控制。程序流程的控制是通过有效的控制结构来实现的, 包括: 分支结构、循环结构。

二、分支结构

分支结构的作用是使程序依照条件而走不同的支路。Visual Basic 支持的分支结构有:

If...Then

If...Then...Else

Select Case

程序中, 往往根据需要进行选择合适的分支结构。

If...Then 语句

用 If...Then 结构可以有条件地执行其结构内的一条或多条语句。

语法格式:

If 条件表达式 Then

多条语句

End If

If...Then...Else 语句

可以使用 If...Then...Else 语句建立多个分支流程, 并根据条件选择其中一个分支。语法结构:

If 条件 1 Then

[语句块 1]