

///

高等学校计算机专业教材

GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI

Windows程序设计

● 管建和 夏军宝 编

GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI
GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI
GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI
GAODENG XUOXIAO JISUANJI ZHUANYE JIAOCAI



人民邮电出版社
POSTS & TELECOMMUNICATIONS PRESS

TP316.7

36

高等学校计算机专业教材

Windows 程序设计

管建和 夏军宝 编

人民邮电出版社

图书在版编目 (CIP) 数据

Windows 程序设计 / 管建和, 夏军宝编. —北京: 人民邮电出版社, 2002.4
高等学校计算机专业教材
ISBN 7-115-09375-X

I. W… II. ①管… ②夏… III. 窗口软件, Windows—程序设计—高等学校—教材
IV. TP316.7

中国版本图书馆 CIP 数据核字 (2002) 第 023207 号

内 容 提 要

本书全面介绍 Windows 操作系统环境下应用程序的开发技术, 以 Visual C++ 开发工具作为主线, 讲解 Windows 应用已成熟的基本编程技术。全书共分十二章, 第 1 章介绍 Windows 编程基础知识; 第 2 章介绍 Windows 窗口命令实现技术; 第 3 章介绍对话框与标准控件的实现技术; 第 4 章介绍 ActiveX 控件的实现和应用技术; 第 5 章介绍 Visual C++ 文档/视结构; 第 6 章介绍图形设备接口 GDI; 第 7 章介绍位图编程技术; 第 8 章介绍多线程编程技术; 第 9 章介绍 MFC 通用类知识; 第 10 章介绍静态库和动态链接库的创建和使用; 第 11 章介绍打印和打印预览功能的实现; 第 12 章简介数据库和 Internet 编程技术。读者通过本书学习, 能够迅速地掌握 Windows 的基本编程方法。本书可作为本科生、研究生学习 Windows 程序设计的教程, 也可作为开发人员的技术参考书。

高等学校计算机专业教材 Windows 程序设计

- ◆ 编 管建和 夏军宝
责任编辑 滑玉
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子函件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
读者热线 010-67180876
北京汉魂图文设计有限公司制作
北京朝阳展望印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 25.25
字数: 616 千字 2002 年 4 月第 1 版
印数: 1.5 000 册 2002 年 4 月北京第 1 次印刷

ISBN 7-115-09375-X/TP · 2266

定价: 32.00 元

本书如有印装质量问题, 请与本社联系 电话: (010) 67129223

出版者的话

为了适应我国大学本科计算机专业教育发展对教材的需要，我社特邀请教育部所属的中国人民大学、中国地质大学、中国农业大学、北京科技大学、北京林业大学、北京语言文化大学（排名不分先后）6所高等学校的“计算机科学与技术系”系主任及资深教授组成专家组，规划、编写、审订了本套教材。

读者对象 本套教材的主要读者对象是普通高等院校计算机科学与技术专业的学生，兼顾信息、自动化及机电类专业学生学习计算机课程的需要。由于这些专业对学生的培养目标是：掌握计算机软件与硬件的基本理论和方法，能从事计算机应用、软件研制、技术开发和管理工作的高级技术人才，因此，本套教材的内容既注意计算机高级技术人才所应具有的知识结构，又适当侧重主要专业知识。

教材特点 本套教材参考了美国 IEEE/CS 和 ACM 技术委员会 2001 年推荐的课程 CC2001(Computing Curricula 2001, 参见 <http://www.csab.org/~csab>, <http://cs.nju.edu.cn/~gchen/teaching/cc2001/overview-bok.html> 2001) 及我国几十所高等院校计算机专业的 2001 年教学计划进行规划。教材内容在选择国际上先进的计算机理论、技术的同时，务求符合我国目前教学环境的实际情况，有一定深度，又有较高的实用性。

根据大学教学的特点，本套教材主要包括必修课程、选修课程和辅助课程三类教材。除了每本教材内容自成体系外，还考虑了它在整个教学计划中的安排顺序，适当增加了承上启下的内容。

写作风格 本套教材根据内容需要，沿着“这是什么、有什么用、怎样用、怎样用得更好”的思路编写教材，通过讲解具体知识，传授学习方法，使学生达到掌握理论和技术的目的；同时力求文笔流畅，言简意赅。

教材每一章除基本知识外，还有本章要点、小结、思考与练习题。有些教材还附加教学大纲（包括教学重点、难点，讲授知识点的参考学时数），操作性较强的课程还配有实验教材（包括上机应具备的软硬件环境和实验内容及方法）。为了方便教师教学，本套教材提供了演示稿，可到人民邮电出版社网站（<http://www.ptpress.com.cn>）的“教材出版图书出版中心”→“教材附件”→“课件”下载。

本套教材的作者都具有多年教学经验，教材的草稿也都在各自的授课过程中多次使用。这套教材的出版，为计算机专业的师生提供了新的选择。我们希望这套以易教、易学，朴实、实用为特色的教材在培养信息化建设专业人才方面做出应有的贡献。

欢迎广大读者对本套教材的不足之处提出批评和建议。

2002 年

编者的话

目前，Windows 操作系统已经是微机上的主流操作系统。Windows 98 与 Windows NT 或更高版本的 32 位 Windows 操作系统平台为应用程序开发人员开辟了一个崭新的编程天地。虽然 Windows 下编程工具较多，诸如 Visual Basic、Dephi、Java 等 RAD 开发工具，而且这些工具易学易用，但作为一个真正的 Windows 程序员还是应该学习 Visual C++。因为 Visual C++ 自诞生以来，一直就是 Windows 环境下最主要的也是最强大的应用开发系统。

实际上，Visual C++ 不仅仅是 C++ 编程语言的集成开发环境，而且与 Win32 紧密相连，因此，我们可以利用 Visual C++ 开发系统完成各种各样应用程序的编写，从底层软件直到上层直接面向用户的软件都可以用 Visual C++ 来完成开发；并且 Visual C++ 强大的调试功能也为大型复杂软件的开发提供了有效的排错手段。随着 Visual C++ 软件版本的不断升级，它的功能也越来越强大，编程也越来越容易，几乎可以实现 Windows 各个方面的应用，Visual C++ 已经成了 Windows 操作系统平台上最强大的应用程序开发工具。

本书从一个程序员的角度系统地介绍了 Visual C++ 6.0 的各项编程技术，向读者揭示了 Visual C++ 6.0 的各项编程技术要点，是一本学习 Visual C++ 6.0 的标准教材。全书着眼于 Windows 下 Visual C++ 已成熟的基本编程技术，力争做到深入浅出、循序渐进地引导学习人员迅速掌握 Windows 下的 Visual C++ 基本编程方法。

本书学习人员只需具备 C++ 编程语言知识并使用过 Windows 环境就可以了。

本书虽然是根据作者长期从事 Windows 编程教学的讲稿和软件开发经验总结而成的，但有些内容是参考了许多资料和网上信息，通过编辑而来的，编写该书，也是作者的一个学习长进过程。基于编者水平有限，书中难免有许多不当之处，敬请大家批评指正。

作者

2002 年 3 月

目 录

第 1 章 引论	1
1.1 Windows 操作系统简介	1
1.2 Windows 图形用户界面	3
1.3 Windows 编程基本约定	6
1.4 SDK 实现的 Windows 应用程序	9
1.5 Visual C++应用程序框架及其特点	18
习题	21
第 2 章 Windows 窗口命令	22
2.1 消息与命令	22
2.2 菜单栏	31
2.3 工具栏	41
2.4 状态栏	45
习题	51
第 3 章 对话框	53
3.1 模式对话框	53
3.2 Windows 标准控件	63
3.3 无模式对话框	93
3.4 系统对话框	99
习题	112
第 4 章 ActiveX 控件	113
4.1 ActiveX 控件简介	113
4.2 控件的实现技术	117
习题	124
第 5 章 文档/视结构	125
5.1 文档、视和应用程序框架	126
5.2 操作事件与输入数据操作	129
5.3 分割器视窗	130
5.4 数据文件的读写	136
5.5 MDI 多类型文档的应用	147

5.6 常见视类介绍	151
习题	152
第 6 章 图形设备接口	153
6.1 设备描述表与图形对象	153
6.2 图形对象介绍	156
6.3 映射方式	169
习题	176
第 7 章 位图	177
7.1 GDI 位图与 DIB 位图	177
7.2 GDI 位图有关函数	182
7.3 位图按钮的实现	196
习题	200
第 8 章 进程与线程	201
8.1 进程与子进程	201
8.2 进程与线程	205
8.3 线程的同步	209
8.4 串行通信与重叠 I/O	213
8.5 一个通信演示程序	218
习题	234
第 9 章 MFC 通用类	235
9.1 字符串类—CString	235
9.2 日期和时间类	241
9.3 集合类的使用	246
9.4 集合类应用的综合实例	254
习题	270
第 10 章 静态链接库和动态链接库	272
10.1 静态链接库	272
10.2 动态链接库	275
10.3 静态链接库和动态链接库的使用	281
习题	291
第 11 章 打印和打印预览	292
11.1 简介	292

11.2 打印和打印预览功能的实现	297
习题	306
第 12 章 高级编程	307
12.1 数据库编程	307
12.2 网络编程	325
习题	336
附录 Visual C++ 6.0 开发环境	337
A.1 Visual C++ 6.0 基本界面	337
A.2 项目与项目工作区	350
A.3 资源及资源管理	354
A.4 使用 AppWizard 建立应用程序	366
A.5 使用 ClassWizard	375
A.6 程序调试	378
A.7 文本编辑器的使用	388
参考文献	394

第 1 章 引 论

早期的 Windows 系统是 Microsoft 公司为 PC 机 MS-DOS 操作系统开发的图形用户界面环境。Windows 系统是一个多任务的操作系统环境。该环境为其应用程序提供统一的窗口和菜单/工具条界面。由于窗口和菜单/工具条界面是统一的，因此对用户来说 Windows 应用程序要比传统的基于 DOS 的程序更易于学习和使用。

1995 年 Microsoft 公司将 Windows 95 以独立系统推出之后，完全改变了 PC 机应用的操作系统环境。新版本一问世，成千上万的将 PC 机操作系统升级为 Windows 95。由于 Windows 95 具有许多新特性，故它不仅给应用计算机带来划时代的变革，也给我们开发人员提供了许多全新的应用程序开发环境。此后，Microsoft 公司又陆续推出了 Windows 98、Windows 2000 等新版本的 32 位操作系统。

1.1 Windows 操作系统简介

大家都知道，Windows 操作系统是美国 Microsoft 公司的产品。追述历史，Windows 系统的第一个版本是 1983 年 12 月首次推出的，其产品命名为 Windows v1.0，当时，它还是一个不成熟的产品，Windows 系统仅仅处于研究阶段，商业化程度较低。后来，Microsoft 公司针对 IBM PC 微型机仅仅只有 256KB RAM、两个软盘驱动器的 8088 硬件系统，重新设计了 Windows 系统，于 1985 年推出了 Windows v1.1 版本，该版本支持窗口的弹出功能。1987 年 10 月，Microsoft 公司发行了 Windows 系统的一个重要版本 Windows v2.0。该版本包括支持重叠式窗口的新用户界面，同时提供支持使用扩展内存功能。但由于 Windows v2.0 只能在实模式下运行，因而能访问的内存的总和要局限到兆字节。

1990 年 5 月，Microsoft 公司发行了一个划时代的版本 Windows v3.0。该 Windows 版本增加了对自画菜单、自画列表框和自画按钮的支持功能。当 Windows v3.0 版在 80386 微机上运行时，可访问 16MB 内存。Windows v3.0 版本很快成为 MS-DOS 机的标准图形用户界面。1992 年 4 月，Microsoft 公司又推出了支持动态数据交换、对象连接和嵌入、多媒体等功能的 Windows v3.1 版本。

1995 年，Microsoft 公司隆重推出了 Windows 95。Windows 95 独立成为微型机的操作系统环境，该系统增加了支持 32 位字长运算的操作系统功能，并也大大增强了用户界面的友好性。Windows 95 保留了对传统的 MS-DOS 的支持，在 Windows 95 操作系统中可以较容易地切换到 MS-DOS，用户还可以像在 MS-DOS 操作系统中工作。1998 年，Microsoft 公司推出了 Windows 98 版本。Windows 98 版本是面向用户的 Windows 操作

系统，与 Windows 95 相比该系统提供了增强的硬件支持、家庭网络功能，改善了联机体验和新的辅助功能。2000 年，Microsoft 公司又推出了 Windows 2000。Windows 2000 提供了几个有用的新辅助工具，如可帮助残疾人快速配置和使用商用计算机，而无需其它软件和硬件等功能。

对用户和程序员来说，Windows 操作系统及其应用程序提供了许多优于其它命令行方式工作的操作系统的特点。用户喜欢诸如“查找和点取”（“find and click”）应用软件、一致的用户界面、多任务机制、与设备无关的图形机制和内存管理的增强机制等等。Windows 操作系统提供的人机交互界面的许多优点是其它操作系统界面无法比拟的。

首先，Windows 系统及 Windows 应用程序的用户可以得到由一致的用户界面所带来的许多认识的方便。因为大多数用户经常使用多种不同的软件，而每一软件都有自己的用户界面，所以用户界面设计的一致性就非常重要。否则，不同的应用程序所使用的击键和热键组合会造成用户使用计算机的混乱。而使用 Windows 系统，每个应用程序窗口包含相同的基本特征，由于这些基本特征在应用程序中是一致的，因此用户可以更容易地适应新应用程序的使用。

其次，Windows 系统用户还受益于使用图像来显示应用程序的数据。由于 Windows 系统是一个图形用户界面，所以，可以用图形/图像形式来表示文件、应用程序、窗口和目录等实际数据信息。用户可以用鼠标通过“单击”点取选择，“双击”点取、拖拉等动作来操作这些物理结构。用鼠标“双击”点取一个图符可以激活相应的应用程序，这比转到相应的目录下，然后输入文件名来激活程序要容易得多。

再者，对用户来说，WYSIWYG（what you see is what you get 翻译为即打即所得）是一个重要优点，例如多数基于文本的文字处理的软件包使用只读存储器及基本输入输出系统（ROM BIOS）字符集来处理显示。由于 ROM BIOS 字符集比例不合适，并与应用程序使用的字体没有直接关系，故在屏幕上显示的文字通常与打印机输出所得到的最终文字不同。然而，有了 WYSIWYG 特性，Windows 能够把文本当作屏幕上的图形图像来处理，因此，这些文本与打印出来的形式是一样的。

另外，Windows 还向用户提供非常重要的多任务功能。由于多种应用程序能在同一时间运行，因而没必要保存正在工作的文件并退出一个应用程序，然后转到另一个应用程序中，用户可继续运行程序并在应用程序之间进行转换。Windows 内存管理能力增强了多任务管理。Windows 为用户提供存取高达几十甚至几百兆扩展内存的功能。Windows 提供卸取所有可用内存资源的功能，因此用户可更容易地优化系统。

Windows 系统的许多优越性可以让用户和程序员共同分享。例如：一致的用户界面（这点对用户很有用）同样能帮助程序员，因为用户界面的基本设计和工具已经确立。从本质上讲，界面对每一种应用程序都是相同的，因而程序员可以把更多的时间花在应用程序功能设计上，而在界面设计上少花一些时间。

Windows 系统图形用户界面的设计是 Windows 系统的又一重要特色。Windows 操作系统允许程序员设计物理结构的图形描述，诸如文件和目录等，并能提供方便的操作机制，例如弹出式菜单和对话框。Windows 系统还支持直接的鼠标和键盘输入，这就可以大大缩短应用程序的开发时间。

程序员将会非常喜欢 Windows 系统的内存管理机制，因为可存取的内存比使用传统的

MS-DOS 应用程序可能存取的内存多得多。而且，在保持系统设计灵活性的同时，程序员可管理大多数系统内存资源。

Windows 系统还提供与开发设备无关的图形的功能。设计好的 Windows 应用程序不必直接访问图形硬件（屏幕和打印机），因而 Windows 应用程序能够使用任何具有 Windows 设备驱动器的显示系统和打印机进行操作。对程序员来说，与设备无关的图形设备意味着代码不依赖于某种特定的系统配置，用 Windows 系统就没有必要为每个应用程序针对所有可能的显示适配器和打印机去开发设备驱动软件。

目前根据处理数据位数的不同，Windows 操作系统主要分为 Win16 和 Win32 两种类型。Microsoft 公司在 1995 年以前推出的 Windows 3.X 版本的操作系统均为 16 位的操作系统，之后推出的 Windows 95、Windows 98、Windows 2000 等后续版本均为 32 位的操作系统。我们这里介绍的 Windows 编程技术主要针对 Win32 操作系统环境。

1.2 Windows 图形用户界面

Windows 应用程序使用窗口在屏幕上进行输入和输出。Windows 应用程序创建应用程序窗口并拥有对窗口的基本访问权，同时应用程序和 Windows 操作系统共同承担管理应用程序窗口的责任。Windows 操作系统负责管理的是应用程序窗口的尺寸、位置和组成，应用程序负责管理应用程序窗口用户区（工作区）。

1.2.1 窗口及其组成

Windows 98 的 Wordpad 写字板应用程序的应用窗口如图 1.1 所示，Wordpad 应用程序窗口是一个典型的 Windows 应用程序窗口。

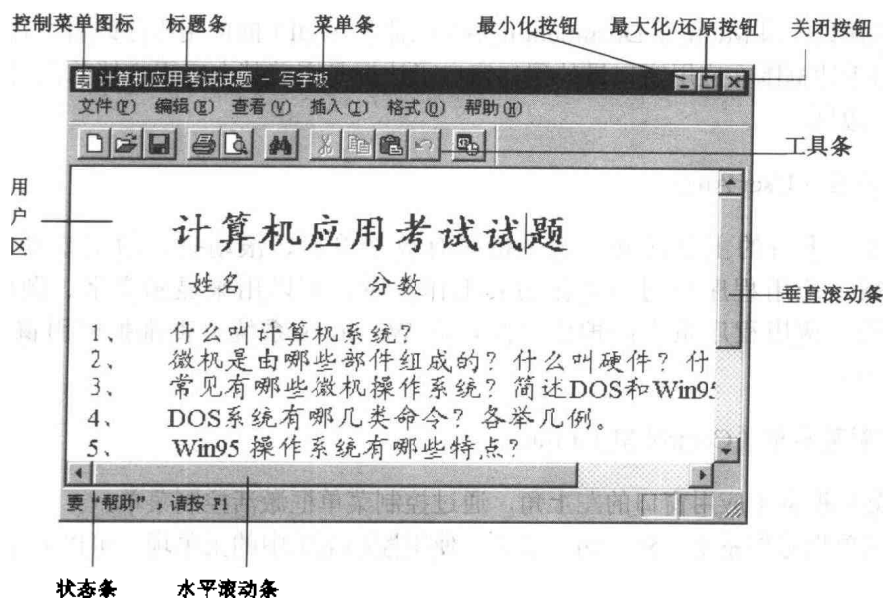


图 1.1 Wordpad 写字板应用程序窗口

通常，一个应用程序窗口主要包括下列组成部分，这些部分可以同时出现，也可以不同时出现在一个窗口中。

- ① 窗口主边框和窗口子边框
- ② 用户区
- ③ 控制菜单框
- ④ 标题条
- ⑤ 菜单条
- ⑥ 工具条
- ⑦ 水平滚动条和垂直滚动条
- ⑧ 最大化/还原窗口按钮
- ⑨ 最小化按钮
- ⑩ 关闭按钮
- ⑪ 状态条

下面简单描述典型的 Windows 应用程序窗口的各个窗口元素。

1. 窗口主边框和窗口子边框 (Main/Child Frame)

单文档接口 (即 Single Document Interface, 简称 SDI) 的应用程序只有一个边框。

边框包括三个基本元素:

- 四个角
- 垂直边
- 水平边

窗口四角能在水平方向和垂直方向上同时改变窗口尺寸。窗口的垂直边框能在水平方向上改变窗口尺寸;窗口的水平边框能在垂直方向上改变窗口尺寸。

多文档接口 (即 Multiple Document Interface, 简称 MDI) 的应用程序既有主边框, 也有子边框。窗口主边框围在应用窗口最外层, 窗口子边框是围着某个应用文档的内层窗口, 它局限于窗口主边框。

2. 用户区 (User Area)

用户区是窗口的主要部分, 它是指没有被菜单栏、滚动条、边框或其它成分占用的窗口区域。应用程序使用户区作为其工作区域, 可以用来显示文字、图形、图像及其它数据等。应用程序负责维护用户区, 而 Windows 系统负责维护应用窗口的位置、大小等成分。

3. 控制菜单框 (Control Menu Bar)

控制菜单框位于应用窗口的左上角, 通过控制菜单框激活控制菜单。

控制菜单也称作系统 (System) 菜单。使用控制菜单中的菜单项, 可以恢复、移动、大小、最小化、最大化及关闭应用窗口。

图 1.2 是 Windows 98 操作系统所带的 Wordpad 应用程序的控制菜单。

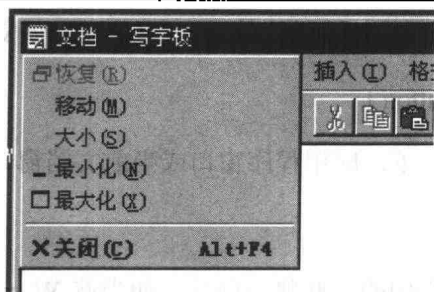


图 1.2 Wordpad 应用程序的控制菜单

4. 标题条 (Title Bar)

标题条列出应用程序的名称，通常用于显示应用程序名或被加工的文件名。活动窗口的标题栏颜色和亮度与非活动窗口不同。

5. 工具条 (Tool Bar)

工具条是用图标按钮形式向用户列出应用程序功能的快速操作方式，其作用类似于菜单条。如提供了一些类似于 File、Edit 和 Help 等功能的工具条按钮，但工具条更直观、更快捷。

6. 菜单条 (Menu Bar)

菜单条列出应用程序向用户提供的所有功能菜单，大多数应用程序都提供 File、Edit 和 Help 菜单，Windows 系统程序员可以为用户设计所需要的功能菜单。

7. 水平滚动条/垂直滚动条 (Horizontal/Vertical Scroll Bar)

如果文本或图像太大，超出了现有的用户区尺寸，可使用水平滚动条移动到文本或图像的任何位置，水平滚动条有三个部分：一是左箭头，在滚动条的最左端。用鼠标在该箭头上点一下，可以使屏幕向左滚动。二是右箭头，在滚动条的最右端，用鼠标在该箭头上点一下，可使屏幕向右滚动。最后一个部分是拇指框，指示当前视口位置。滚动条中填充的部分表示当前视口在全部文本或图像中的位置。

垂直滚动条也有三个部分，一是向上箭头，位于滚动条的最上方，用鼠标在该箭头上点一下，可使文本或图像向下滚动。二是向下箭头，位于滚动条的最下方，用鼠标在该箭头上点一下，可使文件或图像向上滚动。最后是拇指框，指示当前视口位置，滚动条中的填充部分指示当前视口在全部文本或图像中的相对位置。

8. 最大化/还原窗口按钮 (Maximize / Restore Button)

最大按钮可以使应用窗口充满整个屏幕。当用鼠标点取最大按钮后，该按钮就替换成还原窗口按钮 (Restore Button)，用鼠标点取这个按钮可使窗口恢复到原来的尺寸。窗口恢复到原有尺寸后，还原窗口按钮又换成最大化按钮。

9. 最小化按钮 (Minimize Button)

最小化按钮使应用窗口最小化为一个图标。用鼠标在最小化按钮上点一下，应用窗口就

被缩小，在缩小后的窗口（窗口图标）上点一下，可使窗口恢复到原来尺寸。

10. 关闭按钮（Close Button）

用鼠标在关闭按钮上点一下，应用程序窗口或当前文档窗口即被关闭。

11. 状态条（Status Bar）

状态条给出应用程序运行中的一些状态信息，如当前 Word 编辑中的编辑状态信息，当前编辑页面、编辑位置等信息。

1.2.2 Win32 界面元素

Win32 操作系统为我们提供了一组预先构造好的界面元素或控件，用来建立与 GUI 外观和作用相似的应用程序。这种方法使用户具有对 Windows 应用熟悉性，并使用户能较容易地把操作技术的有关知识从熟悉的软件应用传递到新的不熟悉的软件应用中去。我们通过编程还可以自定义控件及式样，给界面设计人员提供创作空间。

Win32 操作系统提供了的丰富 Win32 标准控件，这些控件具有内置式交互作用和应用视觉较美的外观。表 1.1 列出了应用程序中常用的标准控件，并简述了这些控件的基本功能。Win32 里的这些元素实现技术我们将在后面章节中详细讨论。

表 1.1 常用标准控件及说明

标准控件	说 明
按钮	用于启动动作或可选性属性的图形控件
分组框	用于组织其它控件的可视控件
列表框	用于显示项目列表，可以与按钮和文本框相结合
滚动条	用于在水平或垂直方向上滚动窗口
静态文本	用于显示文本或图形元素
编辑框	用于显示和编辑文本
图标列表	图形图标的列表
列表视图	图标和标注的列表
进度指示器	任务完成情况的可视化测量器
RTF 编辑框	字体可格式化的编辑框
标签（Tab）控制	文件夹标签
滑尺	和滚动条相类似的滑尺控制
树型视图	树型结构显示

1.3 Windows 编程基本约定

作为一个 Windows 程序员，不仅要熟悉 Windows 系统操作环境，也要熟悉 Windows 系

统编程环境。到目前为止，我们着重讲解了 Windows 系统操作环境的基础知识，下面我们将介绍 Windows 系统编程环境的一些特点。Windows 系统编程环境通常称为 Windows 应用编程界面 (API)。

一个 Windows 应用软件工程项目通常都是需要多人协作共同开发才能完成的。为方便相互之间引用功能例程，就需要看懂对方的程序，Windows 编程有一些基本约定，下面简述之。

1.3.1 匈牙利表示法

在 Windows 程序设计中，为了提高程序代码的可读性，我们对程序中使用的变量或对象定义其标识符应遵从一定规则。匈牙利表示法就是一种提供定义变量名的规则，它是一种定义变量名的好方法，Windows 程序员经常使用这个方法。匈牙利表示法是以 Microsoft 公司的程序员 Charles Simonyi 的国家命名的，它在 Microsoft 公司编程人员中广泛用于应用程序和系统软件的开发中。

匈牙利表示法的基本方法是用一些小写字母作为变量名前缀，用来指示变量的数据类型，变量名的剩余部分将描述该变量的功能，例如：

nCharacterCounter

指示该变量是一个整数 (n)，表示一个字符计数器 (Character Counter)。

表 1.2 列举了几种 Windows 函数和消息中用作变量和参数名的前缀。

表 1.2 Windows 变量的常用前缀

前 缀	变 量 意 义
b	布尔量 (非 0 是 TRUE, 0 是 FALSE)
c	字符量 (一个字节值)
dw	32 位无符号长型整数
f	用 16 位整数表示的一种位标记
h	16 位句柄
l	32 位长整数
lp	32 位长指针
n	16 位短整数
p	16 位短指针
pt	坐标量 (x,y)，放在 32 位无符号整数中
rgb	RGB 颜色值，放在 32 位整数中

1.3.2 基本数据类型

在用 C 或 C++ 编程时，有的数据类型还挺复杂，为了阅读和书写程序方便，我们通常用 typedef 定义一些容易读和写的等价数据类型。如 LPSTR 等价数据类型为 char*。许多这样的数据类型只是一些基本数据类型的别名，表 1.3 列出了一些在 Windows 编程中常用的基本数据类型的别名。

表 1.3 Windows 基本数据类型

数据类型	对应的基本数据类型	说 明
BOOL	int	布尔值
BSTR	unsigned short *	32 位字符指针
BYTE	unsigned char	8 位无符号整数
COLORREF	unsigned long	用作颜色值的 32 位值
DWORD	unsigned long	32 位无符号整数, 段地址和相关的偏移地址
LONG	long	32 位带符号整数
LPARAM	Long	作为参数传递给窗口过程或回调函数的 32 位值
LPCSTR	const char *	指向字符串常量的 32 位指针
LPSTR	char *	指向字符串的 32 位指针
LPCTSTR	const char * (注 1)	指向可移植为 Unicode 和 DBCS 字符串常量的 32 位指针
LPTSTR	char * (注 1)	指向可移植为 Unicode 和 DBCS 字符串的 32 位指针
LPVOID	void *	指向未定义类型的 32 位指针
LRESULT	Long	来自窗口过程或回调函数的 32 位返回值
UINT	unsigned int	32 位无符号整数
WNDPROC	long (__stdcall *)(void *,unsigned int,unsigned int,long) (注 2)	指向窗口过程的 32 位指针
WORD	unsigned short	16 位无符号整数
WPARAM	unsigned int	当作参数传递给窗口过程或回调函数的 32 位值

注 1: 这是在 DBCS 版本下的情况, 在 Unicode 版本下 LPCTSTR 和 LPTSTR 将代表其它的数据类型。

注 2: 事实上, WNDPROC 被定义为 LRESULT (CALLBACK*)(HWND, UINT, WPARAM, LPARAM), 这个定义最终被编译器解释为 long (__stdcall *)(void *,unsigned int,unsigned int,long)。

1.3.3 句柄

句柄是 Windows 编程的基础, 所以了解其作用非常重要。所谓句柄是 Windows 使用的一种无重复整数, 它主要用来标识应用程序中的一个对象, 如窗口、实例、菜单、内存、输出设备、控制或文件等。例如, 在模块定义文件中, 菜单资源中的菜单项被定义并赋给了一个句柄值。应用程序的菜单条中, 第一个菜单的第一个菜单项可能被赋予 100 这个句柄值, 那么第二项可能被赋予 101 这个句柄值。在应用程序的源代码中, 这些菜单项将要通过 100 和 101 来区分。Windows 应用程序只能访问句柄, 而不能直接访问句柄所指示的实际数据。Windows 系统控制着这些系统数据的存取权, 这样才能在多任务环境中保护这些数据。表 1.4 列举了 Windows 编程中常见的公用句柄类型。

表 1.4 Windows 常见公用句柄类型

句柄类型	说 明
HBITMAP	保存位图信息的内存区域的句柄
HBRUSH	画刷句柄
HCTR	子窗口控件句柄
HCURSOR	鼠标光标句柄
HDC	设备描述表句柄
HDLG	对话框句柄
HFONT	字体句柄
HICON	图标句柄
HINSTANCE	应用程序的实例句柄
HMENU	菜单句柄
HMODULE	模块句柄
HPALETTE	颜色调色板句柄
HPEN	在设备上画图时用于指明线型的笔的句柄
HRGN	剪贴区域句柄
HTASK	独立于已执行任务的句柄
HWND	窗口句柄

我们可以查看 Win32 SDK 文档或者浏览 Windows 头文件（如 `windows.h`、`ctype.h` 以及 `winnt.h` 等）可以获得关于其它数据类型和句柄的有关定义，这些定义往往使用了 `#define` 和 `typedef` 等关键字（见 C 或 C++ 程序设计语言教科书）。

1.4 SDK 实现的 Windows 应用程序

我们学习编写 Windows 操作系统下的应用程序，免不了要了解 Windows API 的有关知识。Windows API（Windows Application Programming Interface，Windows 应用程序编程接口）是 Windows 操作系统给我们开发人员提供的编写应用程序的接口，它是所有 Windows 应用程序开发的基础。所谓 API 就是一系列的例程，我们编写的应用程序都是通过调用这些例程来请求操作系统完成一些低级系统应用服务的。在 Windows 图形用户界面中，应用程序的窗口、图标、菜单、工具条、状态条、消息框和对话框等都是由 Windows API 来管理和维护的。

Windows API 主要依靠三个核心组件，这些组件提供了 Windows 的大部分函数，在 Win16 和 Win32 中，它们具有不同的名称，如表 1.5 所示。