

计算机实用教程

Visual C++

6.0

实用教程

李俊 于良耀
班学钢 黄锦春
编



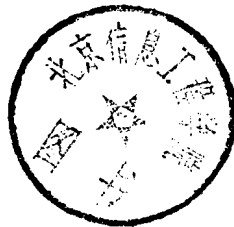
人民邮电出版社



计算机实用教程

Visual C++6.0 实用教程

李俊 于良耀 班学钢 黄锦春 编



人民邮电出版社



7088695

内 容 提 要

JS106/01

Visual C++6.0 是 Microsoft 公司最新推出的功能强大的可视化编程语言, 它可以编写 DOS、Windows 以及 Windows NT 等各种操作系统下的应用程序。

本书不仅是一本全面系统地介绍 Visual C++6.0 编程语言的参考书, 同时对广大的计算机软件人员和爱好者来说也是一本很好的实用教程。书中循序渐进地介绍了 Visual C++6.0 的新增功能, Visual C++ 所特有的消息映射机制、文档与视结构, 各种控件的编程以及利用 Visual C++6.0 进行 ActiveX、数据库和网络编程的各种方法和技巧。阅读本书可以使读者尽快地熟悉和掌握利用 Visual C++6.0 编写各种应用程序的基本知识, 收到事半功倍的学习效果。

本书的读者需要对 Windows 操作系统有一定的了解, 因为它们是利用 Visual C++6.0 编写应用程序时所必需的。

计算机实用教程

Visual C++ 6.0 实用教程

-
- ◆ 编 李 俊 于良耀 班学钢 黄锦春
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
北京朝阳隆昌印刷厂印刷
新华书店总店北京发行所经销
 - ◆ 开本: 787 × 1092 1/16
印张: 31.75
字数: 789 千字
印数: 1 - 6 000 册

2000 年 2 月第 1 版

2000 年 2 月北京第 1 次印刷

ISBN 7-115-08430-0/TP·1559

定价: 46.00 元

编者的话

Visual C++是迄今为止 Microsoft 公司开发的最全面和最完善的程序设计语言。该语言不仅能够编写传统的 DOS 操作系统下的应用程序,更主要的是它可以编写 Windows 操作系统下的各种应用程序。并且它提供了一组各种各样的为适应几乎每一种编程风格而设计的工具。这使得该语言具有功能强大、代码简洁、可移植性强等特点。同时 Visual C++ 语言改变了传统的字符界面的编程手段,利用可视化的编程方法使得先前枯燥、单调的计算机编程变得更加生动和令人赏心悦目。

然而,相对于其他的可视化编程语言,例如 Visual Basic 或者 Delphi 而言, Visual C++ 的内容非常庞大,几乎包括了对计算机所有软件和硬件资源的操作;又由于 Windows 操作系统相对于传统的 DOS 操作系统来说也复杂了许多,所以,要想全面和熟练地掌握 Visual C++ 语言的编程是一件非常费力的事情。尤其是随着计算机和网络技术的飞速发展,在短短的几年中, Visual C++ 就从最初的 1.0 版本上升到了 6.0 版本,其内容也在不断的扩展,特别是在它的后期版本中网络编程的功能得到了大大的增强,这使得对 Visual C++ 语言的学习变得更加困难。因此,我们希望有一本能够全面和详细地介绍最新的 Visual C++6.0 的书籍来帮助广大软件开发人员和爱好者更轻松、更快速地掌握 Visual C++6.0 的编程工作。为此,针对 Visual C++6.0 新增的功能和特点,结合我们在实际编程工作中所积累的一些经验和体会编写了这本书,以便读者能够早日熟悉和掌握这一新的开发工具。

本书共有 21 章,分四个部分对 Visual C++6.0 的功能进行了讲解。结构如下:

第一部分主要介绍 Visual C++6.0 的新增特性,对话框等各种控件、菜单条、状态条的具体内容, Visual C++ 所特有的消息与映射机制、文档与视结构以及在屏幕上绘制图形、打印和打印域览等功能的实现。

第二部分主要介绍了 ActiveX 技术,包括最常使用的 ActiveX 容器、ActiveX 服务器和 ActiveX 控件。

第三部分主要介绍如何利用 Visual C++6.0 编写数据库应用程序,包括使用 Visual C++ 所提供的 MFC 类库编写 ODBC 数据库应用程序,以及利用 Visual C++6.0 新增的 CAO 功能来编写基于 CAO 的数据库应用程序。

第四部分主要介绍了网络编程,内容包括 WinSock 简介、利用 WinInet 类编程、电子函件与浏览器和 ActiveX 控件在网页中的使用。

其中第一部分由李俊编写,第二部分由于良耀编写,第三部分由班学钢编写,第四部分由黄锦春编写。

本书面向的对象是学习 VC 的初级和中级读者,也可供其他设计人员参考。

编者

1999 年 10 月

目 录

第一章 Visual C++6.0 新增特性	1
1.1 本章简介	1
1.2 Visual C++6.0 新增特性和功能	1
1.2.1 Visual C++6.0 概述	1
1.2.2 编译器	3
1.2.3 调试器	3
1.2.4 编辑器	4
1.2.5 项目	6
1.2.6 向导	8
1.2.7 MFC 类库	10
1.2.8 数据库支持	12
1.2.9 例子程序	12
1.2.10 工具	13
1.3 Visual C++ 6.0 的版本介绍	14
1.4 本章小结	14
习题	14
第二章 对话框和控件	15
2.1 本章简介	15
2.2 理解对话框	15
2.2.1 创建对话框资源	16
2.2.2 创建示例对话框	18
2.2.3 建立对话框类	19
2.2.4 使用对话框类	21
2.2.5 无模式对话框	21
2.2.6 显示对话框	22
2.2.7 使用列表框控件	24
2.2.8 使用单选按钮	27
2.3 本章小结	28
习题	28
第三章 Windows 的公共控件	29
3.1 本章简介	29
3.2 控件应用程序	29
3.1 过程状态条控件	31
3.3.1 创建过程状态条	31

3.3.2	初始化过程状态条	32
3.3.3	处理过程状态条	32
3.4	滑块控件	34
3.4.1	创建滑块控件	34
3.4.2	初始化滑块	35
3.4.3	处理滑块控件	35
3.5	Up-Down 控件	36
3.6	图像列表控件	38
3.6.1	创建图像列表	38
3.6.2	初始化图像列表	38
3.7	列表视控件	39
3.7.1	创建列表视	40
3.7.2	创建列表视的栏	43
3.7.3	创建列表视的条目	44
3.7.4	处理列表视	45
3.8	树视控件	46
3.8.1	创建树视	47
3.8.2	创建树视目录	49
3.8.3	处理树视	50
3.9	Rich Edit 控件	50
3.9.1	创建 Rich Edit 控件	51
3.9.2	初始化 Rich Edit 控件	52
3.9.3	处理 Rich Edit 控件	53
3.10	本章小结	56
	习题	56
第四章	消息和命令处理	57
4.1	本章简介	57
4.2	消息队列	57
4.3	消息循环	58
4.4	消息映射	60
4.4.1	消息映射“宏”	60
4.4.2	消息映射如何工作	61
4.5	MFC 处理消息	62
4.5.1	ClassWizard 帮程序员获取消息	62
4.5.5	消息列表	65
4.6	命令	66
4.6.1	命令更新	66
4.6.2	ClassWizard 帮程序员获取命令和命令更新	68

4.7 本章小结	69
习题	69
第五章 文档和视	70
5.1 本章简介	70
5.2 理解文档类	70
5.3 理解视类	73
5.4 创建应用	75
5.5 其他视类	77
5.6 文档模板、视和框架窗口	79
5.7 可持续类和文件 I/O	80
5.7.1 对象和可持续性	80
5.7.2 改进例程序	80
5.8 本章小结	81
习题	81
第六章 在屏幕上绘图	82
6.1 本章简介	82
6.2 理解设备上下文	82
6.3 例程序介绍	83
6.3.1 实现例程序	84
6.3.2 使用字体	86
6.3.3 使用笔	90
6.3.4 使用刷子	91
6.4 滚动窗口	93
6.4.1 创建滚动窗口应用程序	94
6.5 本章小结	98
习题	98
第七章 打印和打印预览	99
7.1 本章简介	99
7.2 理解 MFC 关于打印和打印预览基本功能	99
7.3 多页打印	103
7.3.1 设置每页坐标原点	106
7.3.2 MFC 和打印	108
7.4 本章小结	111
习题	111
第八章 状态条、工具条和菜单条	113
8.1 本章简介	113

8.2	工具条	113
8.2.1	删除工具条	114
8.2.2	添加工具条按钮	115
8.2.3	CToolBar 类的成员函数	117
8.3	状态条	118
8.3.1	创建新命令 ID	119
8.3.2	创建默认的字符串	120
8.3.3	将 ID 添加到指示器数组中	120
8.3.4	创建窗格的命令更新处理函数	121
8.3.5	设置状态条的外观	122
8.4	菜单条	124
8.4.1	添加菜单项	125
8.4.2	添加快捷菜单	128
8.5	本章小结	130
	习题	130
第九章	创建一个完整的应用程序	131
9.1	本章简介	131
9.2	创建显示字符串的例程序	131
9.2.1	用 AppWizard 创建应用程序框架	132
9.2.2	显示字符串	132
9.2.3	添加菜单项	134
9.2.4	创建对话框	135
9.2.5	在 Options 对话框中加入字符串的外观选项	140
9.3	本章小结	144
	习题	144
第十章	帮助文件	146
10.1	本章简介	146
10.2	不同类型的帮助	146
10.2.1	激活帮助	147
10.2.2	帮助的外观	147
10.2.3	使用帮助	149
10.2.4	帮助编程	149
10.3	帮助系统的组成	149
10.3.1	AppWizard 的帮助支持	150
10.3.2	设计你的帮助方式	151
10.3.3	帮助命令的程序实现	151
10.3.4	上下文帮助的编程	152

10.3.5 写帮助文本	155
10.3.6 修改占位字符串	156
10.3.7 增加帮助主题	158
10.3.8 调整目录	162
10.4 本章小结	164
习题	164
第十一章 属性表和向导	165
11.1 本章简介	165
11.2 属性表和属性页类简介	165
11.2.1 类 CPropertySheet	165
11.2.2 类 CPropertyPage	166
11.3 一个简单的属性表例子	167
11.3.1 用 AppWizard 生成程序框架	167
11.3.2 创建两个对话框资源	167
11.3.3 创建 3 个新类	168
11.3.4 修改 CPSheet 类	169
11.3.5 修改主菜单资源	169
11.3.6 为菜单项“属性表”添加处理函数	170
11.3.7 修改 CPropSheetView 类, 响应“属性表”菜单	171
11.4 创建向导的基本知识	173
11.4.1 设置向导模式	173
11.4.2 设定向导页按钮	173
11.4.3 显示向导	173
11.4.4 向导通知消息	173
11.5 向导的例子程序	174
11.5.1 运行 AppWizard 生成程序框架	174
11.5.2 添加对话框资源	174
11.5.3 为三个对话框资源生成类	178
11.5.4 新建 CWizardSheet 类	179
11.5.5 添加用于显示向导的菜单项	180
11.5.6 添加菜单项处理函数	181
11.5.7 显示属性页字符串, 并设置属性页按钮	183
11.5.8 运行程序	186
11.6 本章小结	188
习题	188
第十二章 ActiveX 容器	189
12.1 本章简介	189

12.2	定制 ActiveX 容器的框架	189
12.3	选择 OLE 对象、改变其尺寸和位置	194
12.3.1	增加成员变量 m_rect	194
12.3.2	增加 ItemHitted 函数	195
12.3.3	修改 OnDraw 函数	196
12.3.4	添加一个鼠标单击事件	198
12.3.5	改变光标形状	199
12.4	保存 OLE 对象	200
12.5	复制和粘贴 OLE 对象	201
12.5.1	复制操作	202
12.5.2	粘贴操作	203
12.6	在位激活 OLE 对象	205
12.7	删除 OLE 对象	206
12.8	服务器和容器的参数传递	207
12.9	拖放操作	210
12.9.1	增加必要的变量和成员函数	210
12.9.2	修改 LButtonDown 函数	212
12.9.3	重载类 CContDemoView 的 OnDrop 函数	214
12.9.4	重载类 CContDemoView 的 OnDragEnter 函数	216
12.9.5	重载类 CContDemoView 的 OnDragOver 函数	216
12.9.6	重载类 CContDemoView 的 OnDragLeave 函数	218
12.10	如何插入一个 ActiveX 控件	218
12.11	本章小结	218
	习题	219
第十三章	ActiveX 服务器	220
13.1	本章简介	220
13.2	定制小型服务器	220
13.2.1	生成基本框架	221
13.2.2	分析 MFC 向导生成的代码	223
13.2.3	定制 MiniServer	231
13.2.4	测试	234
13.2.5	其他问题	235
13.3	定制全服务器	236
13.4	定制自动化服务器	237
13.4.1	用 MFC 设计进程内自动化服务器	237
13.4.2	用 MFC 设计进程外自动化服务器	248
13.4.3	用 ATL 设计进程内自动化服务器	253
13.5	本章小结	259

习题	260
第十四章 ActiveX 控件	261
14.1 本章简介	261
14.2 有关 ActiveX 控件的基本概念	261
14.3 使用 MFC 定制 ActiveX 控件	262
14.3.1 形成控件的框架	262
14.3.2 给控件添加属性	265
14.3.3 定制属性页	269
14.3.4 给控件添加方法	270
14.3.5 给控件添加事件	272
14.4 使用 ATL 编写 ActiveX 控件	279
14.4.1 使用 ATL COM AppWizard 创建基本的 ActiveX 控件框架	279
14.4.2 添加属性	282
14.4.3 添加方法	288
14.4.4 添加事件	290
14.4.5 制作属性页	295
14.5 本章小结	302
习题	302
第十五章 ODBC 及 SQL 介绍	303
15.1 本章简介	303
15.2 ODBC 介绍	303
15.2.1 ODBC 的概念	303
15.2.2 ODBC 的组成	304
15.2.3 ODBC 应用程序的工作过程	306
15.3 SQL 介绍	309
15.3.1 SQL 的概念	309
15.3.2 数据定义语言 (DDL)	310
15.3.3 数据控制语言 (DCL)	313
15.3.4 数据处理语言 (DML)	314
15.4 本章小结	328
习题	328
第十六章 利用 MFC 编写 ODBC 程序	329
16.1 本章简介	329
16.2 对数据源的配置	329
16.3 利用 Appwizard 来生成一个数据库应用程序的框架	331
16.4 建立记录集与视图的联系以便显示记录集中的记录	339
16.5 增加, 删除, 更新, 筛选及排序记录集的操作	342

16.5.1 增加及删除一个记录	342
16.5.2 对记录的筛选及排序	352
16.5.3 实现参数化的查询	362
16.6 利用 ClassWizard 来生成自己的记录集类	368
16.7 对多个表的操作	372
16.7.1 实现对两个表的联合操作	373
16.7.2 查询某个部门的平均工资	385
16.8 实现多个记录集与多个记录视	398
16.8.1 利用 ClassWizard 来生成自己的记录视类	398
16.8.2 使用多个记录视窗口	400
16.9 本章小结	408
习题	409
第十七章 利用 DAO 编写数据库程序	410
17.1 本章简介	410
17.2 利用 DAO 来编写自己的数据库应用程序	410
17.2.1 有关 DAO 的基本概念	410
17.2.2 通过 AppWizard 编写 DAO 数据库应用程序	411
17.3 本章小结	419
习题	420
第十八章 用 WinSock 通信	421
18.1 本章简介	421
18.2 基本网络概念	421
18.2.1 客户机/服务器模型	421
18.2.2 套接字概念	422
18.3 提供 WinSock API 低层接口的类 CAsyncSocket	423
18.3.1 初始化套接字	423
18.3.2 创建 CAsyncSocket 对象	424
18.3.3 发送和接收数据报	425
18.3.4 服务器接受连接	426
18.3.5 套接字与服务器连接	427
18.3.6 发送和接收流数据	427
18.3.7 关闭套接字	428
18.4 提供 WinSock API 高层接口的类 CSocket	428
18.5 本章小结	429
习题	429
第十九章 利用 MFC WinInet 类编程	431
19.1 本章简介	431

19.2 WinInet 类介绍	431
19.3 利用 WinInet 类编制客户程序	432
19.3.1 利用 AppWizard 生成程序框架	432
19.3.2 设计对话框	434
19.3.3 添加成员变量	437
19.3.4 查询 Http 和本地文件的处理函数	438
19.3.5 查询 Ftp 目录	443
19.4 本章小结	450
习题	450
第二十章 电子函件和浏览器	451
20.1 本章简介	451
20.2 了解 MAPI	451
20.2.1 客户机应用程序	451
20.2.2 MAPI 分类	452
20.2.3 通用消息调用 CMC	452
20.3 MFC 提供的 MAPI 功能	454
20.4 编写自己的浏览器	457
20.4.1 功能强大的类 CHtmlView	457
20.4.2 编写自己的浏览器	458
20.5 本章小结	470
习题	470
第二十一章 用 ActiveX 控件激活网页	471
21.1 本章简介	471
21.2 动态网页内容	471
21.2.1 静态网页与动态网页	471
21.2.2 ActiveX 控件和 Java Applet 的比较	472
21.2.3 把控件嵌入网页	473
21.2.4 控件的安全性问题	476
21.2.5 用于网页的控件需考虑的问题	480
21.3 利用 AppWizard 创建更快的 ActiveX 控件	481
21.3.1 ActiveX 控件的特征	481
21.3.2 ActiveX 控件的高级特征	482
21.4 本章小结	492
习题	492

第一章 Visual C++6.0 新增特性

1.1 本章简介

Visual C++是当今最流行的开发 32 位应用程序的工具，是体现 Microsoft 公司技术精华的主流产品。它提供了一个由许多组件组成的完整的开发环境，这些组件协同工作，大大简化了软件开发过程。

Visual C++6.0 版是 Microsoft 公司推出的最新版本，较之 Visual C++5.0 版，在很多方面都作了改进。下面先简单地来浏览一下 Visual C++6.0 的新增功能和特性。如果读者曾经使用过 Visual C++以前的版本，那么看完本章会对 Visual C++6.0 有一个大致的了解。

1.2 Visual C++6.0 新增特性和功能

1.2.1 Visual C++6.0 概述

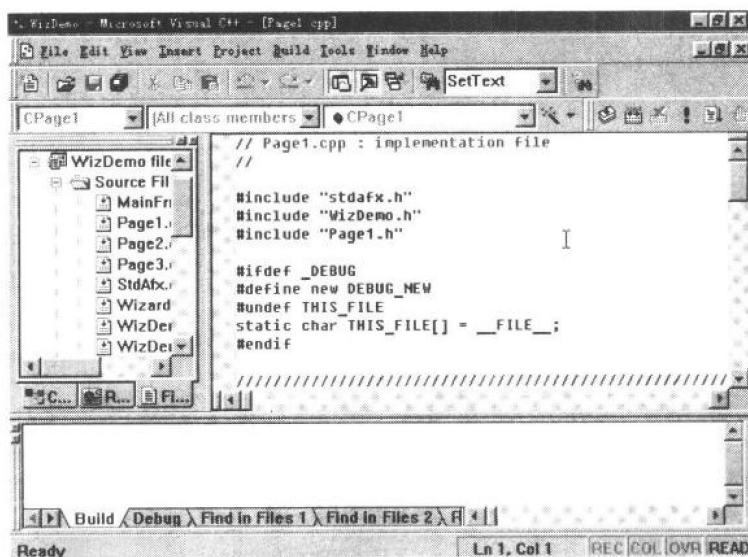


图1-1 Visual C++6.0 界面

首先对比一下 Visual C++5.0 和 Visual C++6.0 的主界面，以对 Visual C++6.0 先有一个感性认识。如图 1-1 所示是读者以后或许将经常面对的 Visual C++6.0 的界面。

图 1-2 是 Visual C++5.0 的界面。如果读者以前使用过 Visual C++5.0 编程的话，一定对此界面非常熟悉。

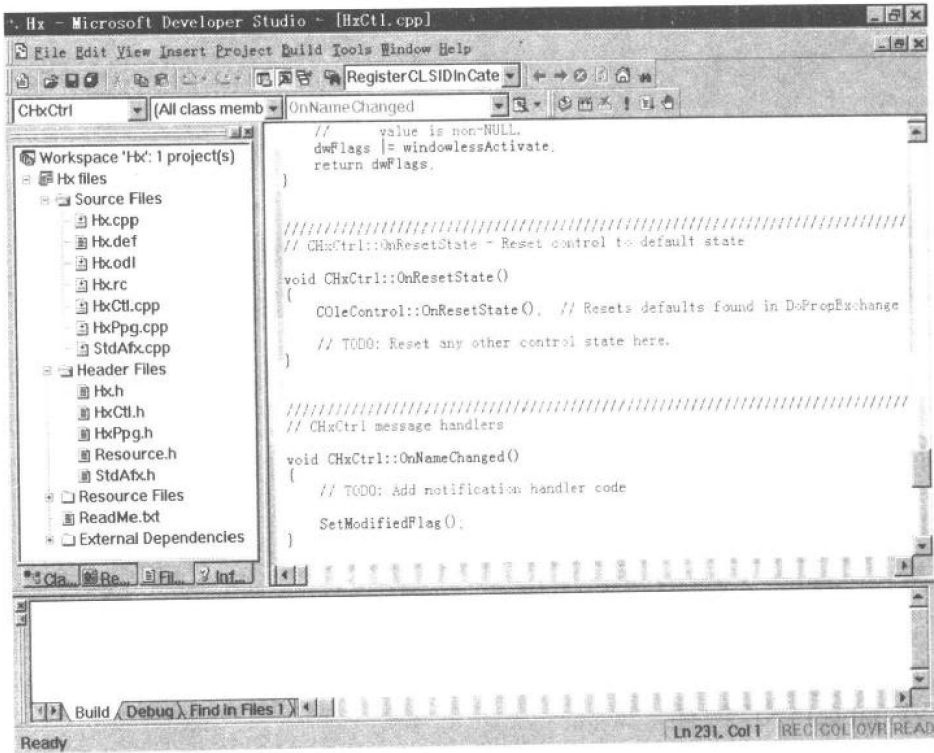


图1-2 Visual C++5.0 界面

大致看一下，读者可能不会觉得很兴奋：VC6 的界面与 VC5 相比没有引人注目的变化，只是在左侧的项目工作区 (Workspace) 窗口中去掉了 InfoView 标签，现在可单击 Standard 工具栏的 Search 按钮来打开 MSDN 帮助系统。以后可以看到，Visual C++6.0 在创建程序的过程、步骤及集成环境的操作上也类似于 Visual C++5.0。然而，Visual C++6.0 在内部当然作出了很多的改进，以适应飞速发展的计算机技术。

微软开发工具的升级总同它的系统平台升级保持同步。Visual C++ 1.52 称为 Windows 3.1 下 16 位应用程序开发的终结产品，Visual C++ 2.0 支持 Windows NT 及 Win32 程序的开发，Visual C++ 4.x 完善了对 Window 95 及 Windows NT 4.0 的支持。当微软在 Windows 95 OSR2 版中捆绑 IE3.0 时，Visual C++5.0 加强了对 Internet 及数据库的支持，提供了更新的完善的 MFC 类库以及 Office 97 式风格的界面，Visual C++5.0 和 Office 97 代表了新一代成熟的 Win95 应用程序。然而使用 Visual C++5.0 开发出的程序却落后于 IE4.0 等产品。在 Windows 98 及 Windows NT5.0 到来后，Visual C++6.0 是弥补这一缺憾的产品。Windows 98 及 Windows NT 5.0 是由 IE4.0 所带动，围绕 Internet 及相关的新技术所诞生的新时期的操作系统。Visual C++6.0 的演化和特性与之相似。

Visual C++6.0 在两个方向上进行了提高和改进：一是纵向的、随系统平台进行的技术升级，如：支持 IE 4.0 的控件、完善对 OLE DB 的支持、ADO、动态 HTML、HTML 帮助系统以及类似于 Explorer 的应用程序和 Office 97 形式的工具条和界面；另一方面是横向的、Visual studio 开发平台内部工具之间集成化的调整和提高，如：帮助系统的独立化使 Visual Studio 各部分使用统一的帮助系统；Visual Modeler 提供可视化的 CASE 工具，使大型面向对象的软件工程得以实现，可以使用同样的模型进行 VC、VB 的开发；Repository 实现跨应用的对象共享，使对象重用技术更真实、有效。

Visual C++6.0 的集成环境虽然在外观上没有什么变化，但内部却有许多重大的改进和新的特性。首先来浏览一下其中比较重要的一些改进。

1.2.2 编译器

下列这些编译器特征是 Visual C++ 6.0 新增的。

1. 更好的内联 (Inline) 控制

为了更好地实现内联控制，现在有三个关键词可用：`inline__inline` 和 `__forceinline`。`inline` 和 `__inline` 关键词告诉编译器试着使用内联函数，优化器将根据优化开关（基于大小还是速度）来决定是否采用内联函数。为了使编程者更好地控制内联函数的使用，Visual C++6.0 支持 `__forceinline` 关键词。这个关键词告诉编译器强制使用内联函数，除非在某些情况下不可能采用内联函数。

2. 依赖性 (Dependencies) 用文本文件存储

输出的 `make` 文件现在把文件依赖性保存在一个独立的可编辑的文本文件中。

3. 运行错误检查

Visual C++编译器能在程序中植入某些运行错误检查代码。编译器的选项 `/GZ` 使这些检查能被执行。运行错误检查只在调试 (`debug`，利用 `/Od` 选项编译)项目时使用。这些新的检查用于局部变量自动初始化、函数指针调用栈验证和调用栈验证。

1.2.3 调试器

下列这些调试器特征是 Visual C++ 6.0 新增的。

1. 用于应用程序诊断的函数 `AfxDumpStack`

`AfxDumpStack()` 函数生成当前栈的一个映像，并把此映像输出到调试输出设备（如 `afxDump`）。这给你提供了一个得到应用程序诊断信息的机制，即使用户机器上没有安装调试器。它可用于 MFC 类库的调试和非调试版本。

2. 编辑和继续

编辑和继续允许你调试时进行简单的编辑，但不必退出调试阶段、重编译连接然后重新进入调试器。编辑改动将被重新编译，并在内存中添加到执行的应用程序中去。

3. 解码显示 GUIDs

基于 GUIDs 的类型（包括 IIDs, CLSIDs, 和 REFIIDs）用名字来显示（如果在注册表中找到）；如果注册表中没找到，则用通常的 GUID 十六进制形式显示。

4. 改进的虚拟表 (V-Table) 和函数指针显示

只要可能, 虚拟表入口和函数指针现在将用符号显示。而在 Visual C++ 5.0 中, 这些项显示为十六进制的地址值。

5. 显示 MMX 寄存器

通过使用符号 MM0-MM7, 你可以在 Watch 窗口和 Quickwatch 窗口显示 MMX 寄存器。MMX 寄存器是 64bit 的整型寄存器, 不管它们是否支持 MMX 指令, 都以 x86 机器中的形式显示。

6. Module 列表对话框

这个新的对话框显示你程序中所有 DLLs 的地址、名字、路径和装载顺序。你可以利用此对话框来知道程序现在正使用哪个 DLL。

1.2.4 编辑器

这些编辑特性是 Visual C++ 6.0 新增的。

1. 自动补全声明

文本编辑器现在支持自动补全声明的特征, 可显示类的成员、函数原型、标识符声明, 以及在鼠标或光标位置处显示代码注释 (如图 1-3 所示)。

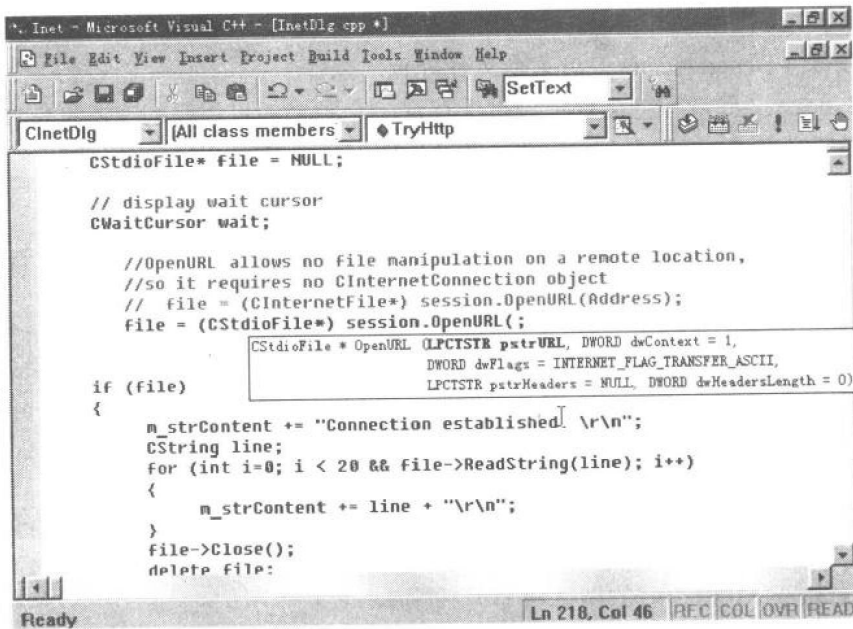


图1-3 输入时提示函数原型

只要在弹出的快捷窗口中找到你需要的代码元素, 简单地敲一下回车, 就可以把其直接插入你的源代码 (如图 1-4 所示)。这样相当于把 MFC、Win32 和 ATL 库方便地放在了你的手边, 成员变量和函数原型很清楚; 还可以减少一些打字的负担, 比如有些类或成员的名字很长。