

Visual

C++ 6.0

开发技巧 及实例剖析

齐舒创作室 编著 刘 鎏 审校



清华大学出版社

<http://www.tup.tsinghua.edu.cn>

TP312C

91



Visual C++ 6.0

开发技巧及实例剖析

91.12

齐舒创作室 编著

刘 鎏 审校

清华大学出版社

(京)新登字 158 号

内 容 摘 要

本书用丰富的实例向读者展示了如何利用 Visual C++ 6.0 编写应用程序。全书共分四个部分。第一部分为基础篇,主要讲述如何利用 Visual C++ 编写一些常见的用户界面元素;第二部分为提高篇,主要引导读者从操作系统的角度去理解应用程序;第三部分、第四部分为两个专题篇,主要介绍当今比较热门的 ActiveX 技术和数据库技术,使读者了解这两门技术,并为进一步提高打下基础。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标签,无标签者不得销售。

图书在版编目(CIP)数据

Visual C++ 6.0 开发技巧及实例剖析/齐舒创作室编著. —北京:清华大学出版社,1999.8
ISBN 7-302-03681-0

I. V… II. 齐… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(1999)第 34948 号

出版者:清华大学出版社(北京清华大学校内,邮编 100084)

印刷者:北京市清华园胶印厂

发行者:新华书店总店北京发行所

开 本:787×1092 1/16 印张:36 字数:872

版 次:1999 年 11 月第 1 版 1999 年 11 月第 1 次印刷

书 号:ISBN 7-302-03681-0/TP·2049

印 数:0001~6000

定 价:59.00 元

前 言

可视化编程技术是当今最流行的编程技术。在众多的可视化编程软件中,Microsoft 公司的 Visual C++ 独树一帜。它以其方便易用的 AppWizard、ClassWizard 以及丰富的可视化资源编辑工具而著称,特别适用于开发一些要求调整的大型应用软件。此外,Visual C++ 还提供一些诸如 OpenGL、DirectX、DAO 之类的专业库,这更进一步增强了 Visual C++ 的功能。

本书是针对如何使用 Visual C++ 而编写的。全书共分为四个部分。第一部分讲述了 Visual C++ 编程的一些基础知识,主要讲一些用户界面元素的编制和管理方法;第二部分是编程进阶,主要讲述内存管理、多任务编程以及动态链接库;第三部分、第四部分是两个专题篇,分别讲述了 ActiveX 技术和数据库编程技术。

本书在编写过程中,力求从易学性、实用性两个角度出发。在每一个新概念出现时,先讲述一些必要的基础知识,之后介绍一些常用函数的用法,再给出实例,一步一步引导读者完成。在完成实例后,再对某些比较难以理解的程序代码给予分析说明。此外,为了让用户编写出一些实用的程序,本书还例举了一些较难的示例。这些示例不适于初学者,初学者可以先跳过这些示例,在确实需要时再进一步学习。对于这些例子,书中有所提示。

本书可作为 Visual C++ 入门的基础教材提供给初学者。由于书中有一些实用的代码段,对某些编程方法及过程也作了总结,所以便于读者查询使用。因而,本书是 Visual C++ 初学者以及编程爱好者不可多得的参考书。

在软件蓬勃发展的今天,早日掌握领导世界软件发展潮流的 Microsoft 公司的 Visual C++ 必定能增强你在软件编程方面的竞争力。在这里,作者祝读者编程愉快。

本书是集体智慧的结晶,但由于时间短促,再加之作者的水平有限,书中的错误以及疏漏在所难免,望广大专家、同行批评指正。

作者

1999 年 2 月

目 录

第一部分 Visual C++ 6.0 基础编程

第 1 章 Visual C++ 6.0 概述与编程基础	2
1.1 Visual C++ 6.0 的新特征	2
1.2 Visual C++ 6.0 的主窗口	4
1.3 Visual C++ 6.0 的系统工具	5
1.3.1 菜单编辑器	5
1.3.2 加速键编辑器	8
1.3.3 资源符号	10
1.3.4 图形编辑器	11
1.3.5 工具栏编辑器	13
1.3.6 对话框编辑器	14
1.3.7 串编辑器	17
1.4 MfcApp 应用程序	18
1.5 ConsoleApp 应用程序	28
1.6 本章小结	31
第 2 章 图形设备接口	32
2.1 图形设备接口简介	32
2.2 设备描述表	33
2.3 坐标映射	35
2.3.1 设置映射模式	35
2.3.2 定义窗口的范围	36
2.3.3 定义视图区	36
2.3.4 设置视图区的原点	37
2.4 绘制基本图形	37
2.4.1 画点	37
2.4.2 画直线	38
2.4.3 画折线和曲线	39
2.4.4 画封闭曲线	41
2.5 画笔	49
2.5.1 创建画笔	49
2.5.2 画笔的属性	49
2.5.3 库存画笔	50
2.6 画刷	57
2.6.1 纯色画刷	57

2.6.2 阴影画刷	58
2.7 字体	64
2.7.1 显示文本	64
2.7.2 设置文本颜色	64
2.7.3 设置字符间距	65
2.7.4 设置文本对齐方式	65
2.7.5 得到文本度量	66
2.7.6 字体的创建	68
2.8 本章小结	73
第3章 菜单	74
3.1 编辑菜单资源	74
3.2 菜单消息响应	75
3.3 菜单的 UI 机制及显示状态的修改	78
3.4 动态地改变菜单	79
3.4.1 获得菜单的指针	79
3.4.2 给菜单中增加菜单项	80
3.4.3 删除菜单项	80
3.4.4 获得菜单的菜单项数	80
3.4.5 使菜单项有效或无效	81
3.5 上下文菜单	99
3.6 本章小结	103
第4章 系统的输入设备	104
4.1 鼠标	104
4.1.1 鼠标消息	104
4.1.2 鼠标消息处理	106
4.1.3 改变鼠标指针的形状	107
4.1.4 捕捉鼠标	108
4.2 键盘	115
4.2.1 键盘消息	115
4.2.2 字符消息	118
4.3 本章小结	125
第5章 工具栏和状态栏	126
5.1 工具栏	126
5.1.1 工具栏资源	127
5.1.2 按钮的状态	128
5.1.3 工具栏类	128
5.1.4 消息传递机制	130

5.1.5 普通工具栏示例	131
5.1.6 高级工具栏示例	137
5.2 状态栏	150
5.2.1 CStatusBar 类	150
5.2.2 状态栏的创建和处理	151
5.2.3 状态栏与提示信息	151
5.2.4 状态指示器	153
5.2.5 状态栏示例一	154
5.2.6 状态栏示例二	156
5.3 本章小结	159
第 6 章 MFC 异常处理	161
6.1 Visual C++ 的异常处理机制	161
6.2 设置自己的 terminate() 函数处理器	162
6.3 MFC 中的异常类	163
6.3.1 CMemoryException	163
6.3.2 CNotSupportedException	164
6.3.3 CFileException	164
6.3.4 CArchiveException	165
6.3.5 CResourceException	165
6.3.6 COleException	165
6.3.7 CUserException	166
6.3.8 CDBException	166
6.3.9 CDaoException	166
6.3.10 CInternetException	166
6.3.11 Win 32 系统级异常	166
6.4 本章小结	168
第 7 章 对话框和控件	169
7.1 对话框资源	169
7.2 对话框中的控件	170
7.2.1 CListBox	171
7.2.2 CComboBox	171
7.2.3 CStatic	171
7.2.4 CButton	171
7.2.5 CBitmapButton	171
7.2.6 CEdit	172
7.2.7 CScrollBar	172
7.3 CDialog 类与对话过程函数调用顺序	172

7.4	DDX 和 DDV	174
7.5	预制对话框	175
7.6	模式对话框示例	183
7.7	无模式对话框示例	193
7.8	基于对话框的应用程序	207
7.9	本章小结	219
第 8 章	切分窗口、文档和多视图	220
8.1	切分窗口	220
8.2	CSplitterWnd 类	220
8.3	切分窗口与多视图	221
8.4	单文档动态切分窗口示例	222
8.5	单文档静态切分窗口示例	225
8.6	文档和视图	231
8.6.1	CDocument 类	231
8.6.2	CView 类	232
8.6.3	文档—视图之间的相互作用函数	233
8.6.4	文档模板	235
8.7	多文档界面	237
8.8	动态切分的 MDI 应用程序	237
8.9	具有多个视图类、不带切分窗口的 MDI 应用程序	239
8.10	静态切分的 MDI 应用程序	252
8.11	本章小结	255

第二部分 Visual C++ 6.0 编程进阶

第 9 章	内存管理技术	258
9.1	内存管理的发展	258
9.2	简单内存分配	259
9.2.1	栈结构分配	259
9.2.2	堆分配	259
9.3	Windows 中的内存分配函数	261
9.4	分配可移动内存和可放弃内存	262
9.5	虚拟内存	264
9.6	共享内存	264
9.7	本章小结	267
第 10 章	多任务编程	268
10.1	进程、线程和优先级	268
10.2	工作线程和用户界面线程的创建	270

10.2.1	工作线程	270
10.2.2	使用工作线程	272
10.2.3	创建用户界面线程(UI)	277
10.3	线程的挂起及唤醒	278
10.4	使线程休眠	278
10.5	线程同步	279
10.5.1	使用事件对象	279
10.5.2	使用临界段	291
10.5.3	使用互斥量	294
10.5.4	使用信号量	298
10.6	线程的终止	304
10.7	如何编写基于进程的多任务操作	307
10.8	本章小结	310
第 11 章	动态链接库	311
11.1	动态链接库	311
11.1.1	动态链接和静态链接	311
11.1.2	创建动态链接库的原因	311
11.1.3	动态链接库的基础	312
11.2	普通的 DLL	313
11.2.1	理解普通的 DLL 本质	313
11.2.2	动态加载 DLL	315
11.2.3	实现 TestDll 应用程序	315
11.3	扩展 MFC DLL	331
11.4	本章小结	333

第三部分 ActiveX 编程

第 12 章	容器应用程序	336
12.1	创建框架容器应用程序	336
12.2	管理嵌入对象的大小和位置	342
12.3	使用鼠标选取项	344
12.4	本章小结	370
第 13 章	服务器应用程序	371
13.1	创建框架服务器应用程序	371
13.2	定制应用程序的资源	377
13.3	完善文档类和视图类	379
13.4	完善服务器项	388
13.5	运行 Server 应用程序	394

13.6 本章小结	399
第 14 章 自动化	400
14.1 自动化服务器	400
14.1.1 完善框架自动化服务器应用程序	400
14.1.2 定制自动化服务器的资源	405
14.1.3 完善文档类和视图类	405
14.1.4 定义服务器的属性和方法	408
14.2 自动化客户应用程序	418
14.2.1 完善框架自动化客户应用程序	418
14.2.2 定制自动化客户的资源	422
14.2.3 从类型库中创建类	423
14.2.4 完善客户应用程序的视图类	424
14.3 本章小结	433
第 15 章 ActiveX 控件	434
15.1 ActiveX/OLE 控件	434
15.2 ActiveX 控件的属性、事件与方法	434
15.2.1 属性	434
15.2.2 事件	435
15.2.3 方法	436
15.3 ActiveX 控件示例	436
15.3.1 创建框架 ActiveX 控件	436
15.3.2 初始化控件的大小以及在控件中显示当前时间	439
15.3.3 使控件动态显示当前时间	441
15.3.4 增加控件属性	442
15.3.5 增加标准事件	444
15.3.6 增加定制事件	446
15.3.7 增加控件方法	447
15.4 本章小结	463

第四部分 数据库编程

第 16 章 ODBC	466
16.1 ODBC 简介	466
16.1.1 常用术语	466
16.1.2 ODBC 与标准化	466
16.2 SQL 语法基础	469
16.2.1 结构化查询语言	469
16.2.2 关系对象	470

16.2.3	SQL 分类	471
16.2.4	SQL 语法	471
16.3	ODBC 的实现	476
16.3.1	配置 ODBC 环境	476
16.3.2	连接数据源	476
16.3.3	查询数据和数据源	479
16.3.4	装备和执行 SQL 请求	479
16.3.5	检索数据	480
16.3.6	释放 ODBC 环境	481
16.4	ODBC 的应用	481
16.4.1	数据库的创建	482
16.4.2	配置 ODBC 环境	486
16.4.3	创建 ODBC 应用程序	488
16.5	操作数据库的内容	496
16.5.1	给表增加一个记录	496
16.5.2	如何删除记录	500
16.5.3	在数据库中查找数据	501
16.6	本章小结	512
第 17 章	MFC 数据库类	513
17.1	MFC 数据库类	513
17.2	CDatabase 类	513
17.2.1	打开一个数据库连接	513
17.2.2	连接信息	516
17.2.3	用 CDatabase 执行 SQL 语句	516
17.2.4	用 CDatabase 进行事务处理	516
17.3	CRecordset	518
17.3.1	构造一个记录集	520
17.3.2	使用 RFX 读写数据	520
17.3.3	刷新记录集	521
17.3.4	关闭记录集	522
17.3.5	在记录集中移动	522
17.3.6	成批取回行	523
17.3.7	改变记录集中的数据	527
17.3.8	事务处理	528
17.3.9	使用语句参数	528
17.3.10	记录集错误处理	529
17.4	CRecordView 类	529

17.4.1 创建一个 CRecordView	530
17.4.2 用 CRecordView 进行对话框数据交换	530
17.4.3 OnGetRecordset()函数	530
17.4.4 CRecordView::OnMove()函数	530
17.5 本章小结	531
第 18 章 DAO 编程	532
18.1 DAO	532
18.2 ODBC 与 DAO 的选择	532
18.3 使用 DAO 构造数据库应用程序	533
18.3.1 创建一个简单的网格视图应用程序	534
18.3.2 创建和运行应用程序	541
18.3.3 给应用程序增加报表	547
18.4 本章小结	561
附:CD-ROM 说明	562

第一部分

Visual C++ 6.0 基础编程

在这一部分中主要向读者介绍用 Visual C++ 编写应用程序的一些基础知识,它包括第 1 章到第 8 章的内容。

通过学习这一部分,读者便能够编写一些使用常用用户界面元素的程序,可以通过菜单、工具栏、状态栏、对话框在应用程序与用户之间进行交互,可以利用图形设备接口向用户反映文档中的数据,能够理解切分窗口与视图以及文档与视图之间的关系。此外,在本部分中还向读者简单介绍了异常处理。

总之,作为 Visual C++ 编程的基础,本部分充分考虑了读者的易学性与实用性。在举例时,力求两者兼顾,同一类应用程序一般都有两个示例,第一个较简单易学,第二个较复杂实用。

第1章

Visual C++ 6.0 概述与编程基础

Visual C++ 是在 Microsoft C 的基础上发展起来的。随着计算机软、硬件技术的飞速发展,如今 Visual C++ 已成为集编辑、编译、运行、调试为一体的功能强大的集成编程环境。在可视化编程技术如火如荼的今天,Visual C++ 可谓独占鳌头,它为大型工程项目的开发提供了强有力的工具。

本章是为了让初学者快速熟悉 Visual C++ 编程环境而设定的。读者在学完本章后,能够具备 Visual C++ 编程的基本概念。

1.1 Visual C++ 6.0 的新特征

与 Visual C++ 5.0 相比较,Visual C++ 6.0 增加了许多新特征。这些新特征在编辑器、编译器、调试器中都有所反映。下面我们将介绍一些在编程过程中经常用到的新特征。

1. 编辑器的语句自动完成特征

这种特征是由 Intellisense 系统实现的。当用户编辑代码时,Intellisense 会在光标或鼠标位置附近显示类的成员函数、变量以及全局变量,如图 1.1 所示。用户可以在成员列表中选

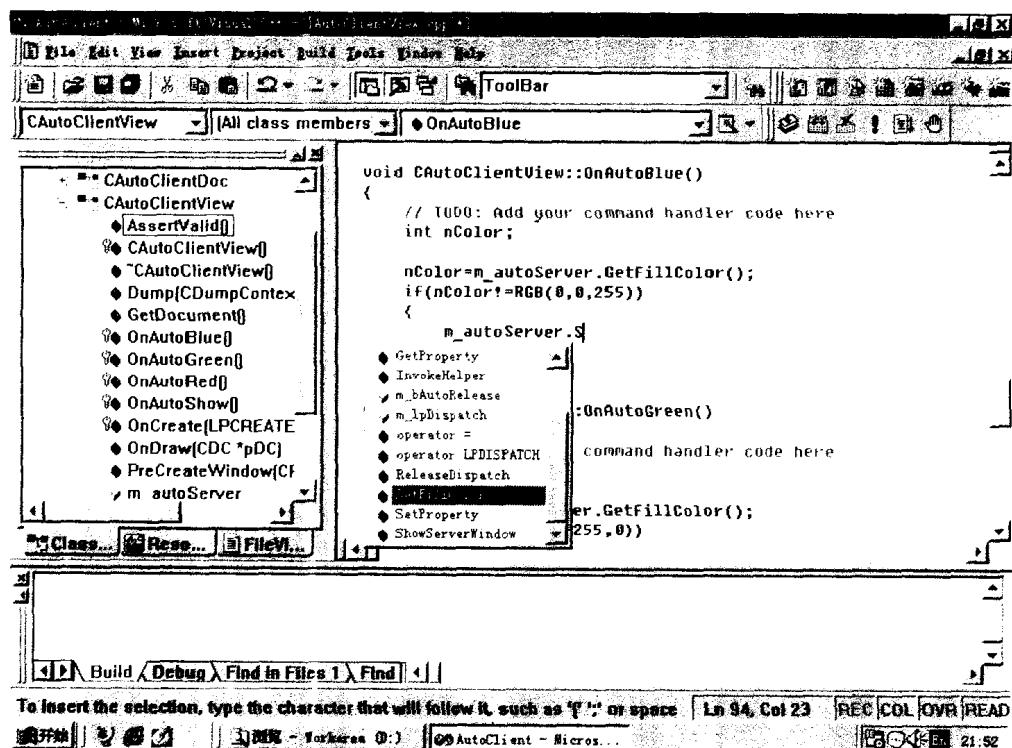


图 1.1 Intellisense 自动显示类的成员函数、变量

择成员来插入代码中。

使用 Intellisense, 用户还可以查看代码注释、函数声明和变量类型等信息, 如图 1.2 所示。

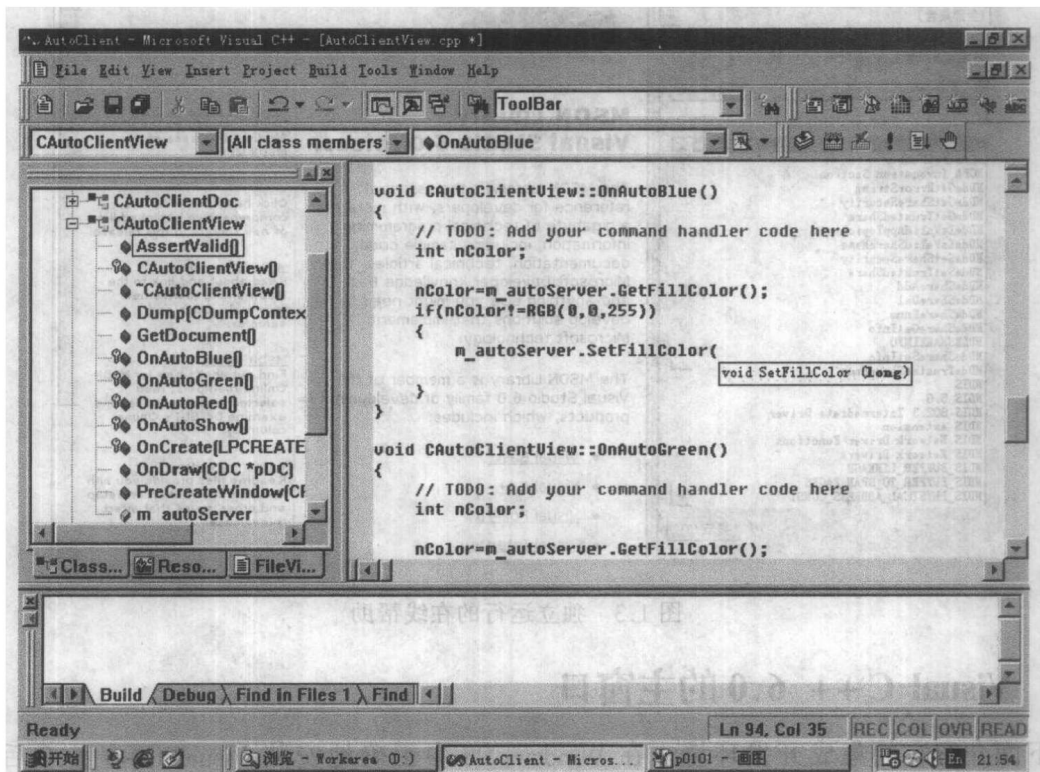


图 1.2 Intellisense 自动显示函数原型、标识符定义等

Intellisense 的存在使工作更轻松, 编写的代码更不容易出错。

2. AppWizard 的增强

在 Visual C++ 6.0 中, 当用户使用 MFC AppWizard 来创建应用程序时, AppWizard 有了更多的选项。现在 (Visual C++ 6.0) 的 AppWizard 可以支持不基于文档/视图结构的单文档界面或多文档界面应用程序。此外, AppWizard 还可以创建具备 Windows 资源管理器风格、Web 浏览器风格的应用程序。

3. 在线帮助与集成编程环境相分离

当打开 Visual C++ 6.0 帮助时, 可以发现它与 Visual C++ 5.0 之间的不同。首先, Visual C++ 5.0 是把帮助文件集成到 Developer Studio 中的, 而在 Visual C++ 6.0 中, 帮助是一个独立的应用程序, 可以单独运行。用户在集成开发环境中调用帮助时, 系统会自动打开这个应用程序, 如图 1.3 所示。

另外, 从上图中可以看到, Visual C++ 6.0 的帮助是使用 HTML 语言编写的, 这使得访问相关帮助时更简单方便。

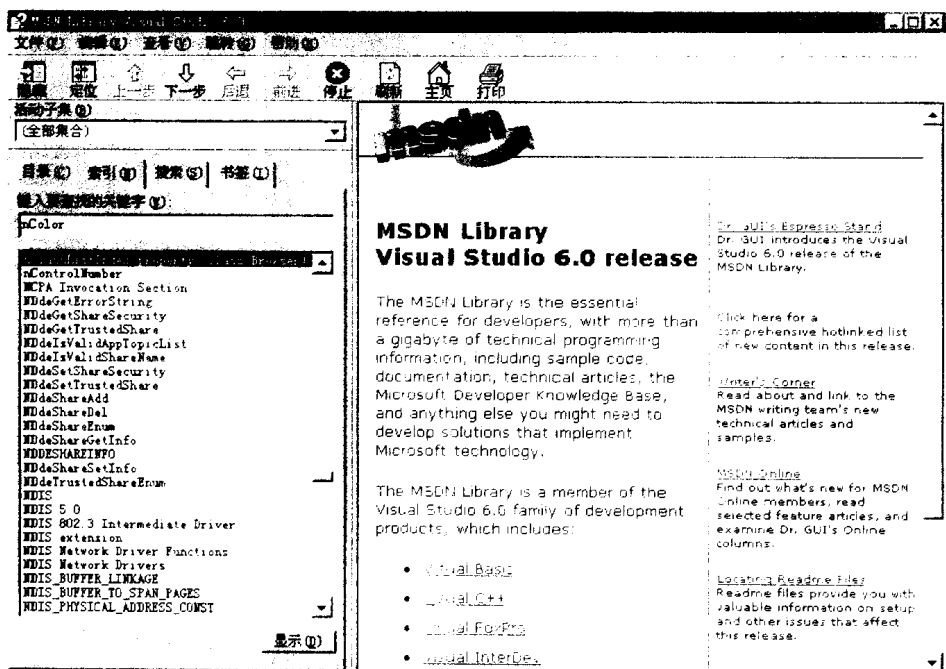


图 1.3 独立运行的在线帮助

1.2 Visual C++ 6.0 的主窗口

下面介绍它的集成开发环境。在 Windows 98 中正确地安装 Visual C++ 6.0 后,单击【开始】按钮,从开始菜单中启动 Visual C++ 6.0,进入开发环境 Developer Studio。打开一个项目后,我们可以看到 Developer Studio 由标题栏、工具栏、工作区窗口、源代码编辑窗口、输出窗口和状态栏组成(见图 1.4 所示)。

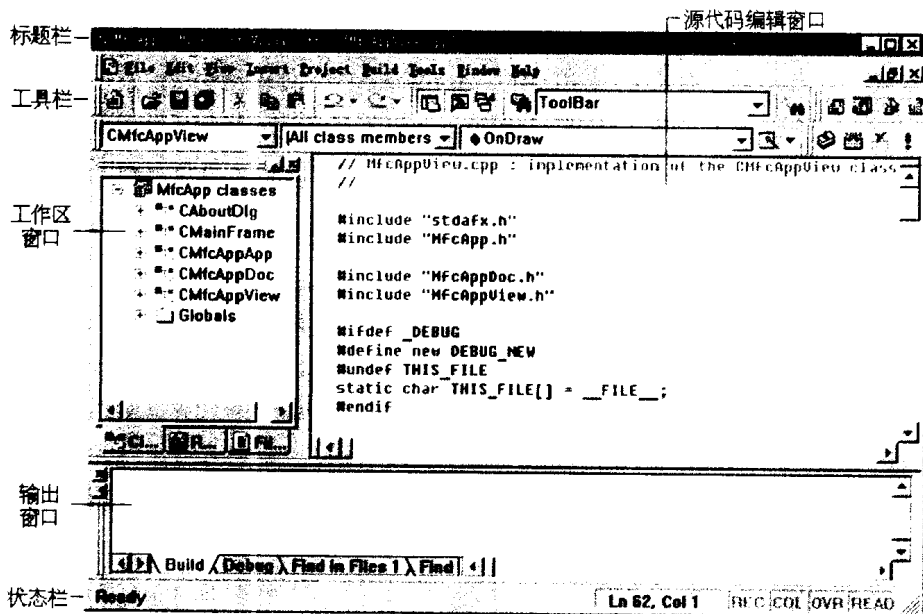


图 1.4 Visual C++ 6.0 开发环境

屏幕最上端是标题栏。标题栏用于显示应用程序名和所打开的文件名,标题栏的颜色可以表明对应窗口是否被激活。标题栏左端为控制菜单框,它是用于打开窗口控制菜单的图标。用鼠标单击该图标,系统将弹出窗口控制菜单。窗口控制菜单用于控制窗口的大小和位置,如还原、移动、关闭、最大化和最小化等。标题栏的右边有三个控制按钮,从左至右分别为最小化按钮、还原按钮和关闭按钮,这些按钮用于快速设置窗口大小,例如,使窗口填充整个屏幕、将窗口最小化为图标或关闭窗口。

标题栏的下面是菜单栏和工具栏。工具栏的下面是两个窗口,一个是工作区窗口,一个是源代码编辑窗口。工作区窗口的下面是输出窗口,它用于显示项目建立过程中所产生的各种信息。屏幕最底端是状态栏,它给出当前操作或所选择命令的提示信息。

1.3 Visual C++ 6.0 的系统工具

为支持可视化编程,Visual C++ 6.0 除了提供集成的编译器、调试器以外,还提供了大量的用于可视化编程的资源编辑工具。这些资源编辑工具主要有菜单编辑器、加速键编辑器、资源符号编辑器、图形编辑器、工具栏编辑器、对话框编辑器等编辑器。这些编辑器对于可视化编程具有重要作用。

1.3.1 菜单编辑器

菜单编辑器用于创建并编辑菜单资源。使用菜单编辑器,我们可以创建标准菜单和菜单选项,可以为菜单或菜单选项定义热键、加速键和状态栏提示;也可以创建上下文菜单,以使用鼠标右键来执行要频繁使用的命令。建立菜单或菜单选项后,可以用 ClassWizard 为菜单选项编写要执行的代码。图 1.5 是打开某一菜单资源后的菜单编辑器。

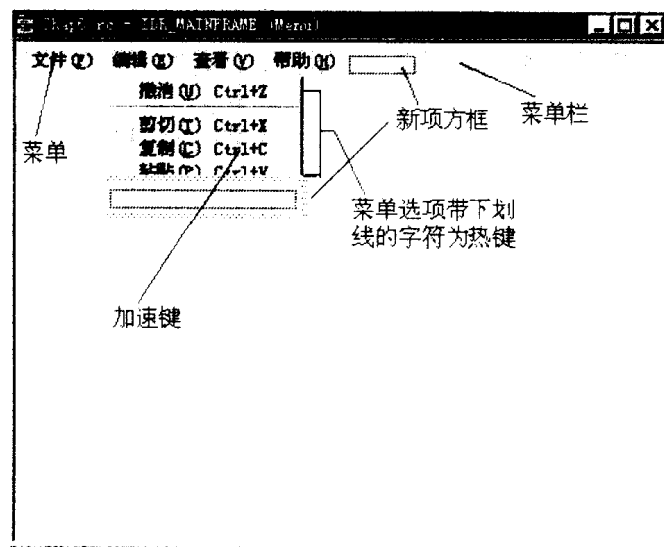


图 1.5 菜单编辑器

1. 创建菜单和菜单选项

进入菜单编辑器后,就可以在菜单栏中创建菜单和菜单选项。