



流行编程软件简明教程丛书

五洋创作

# Visual C++ 简明教程

邓 力 编著

张 晋 审校

● 中国大地出版社 ●  
2001 年 · 北京



## 内 容 提 要

Visual C++是 Windows 平台上使用最广泛的可视化开发环境，功能非常强大。本书共十章，简明扼要地介绍了如何使用 Visual C++开发 Windows 应用程序，主要内容包括：MFC 的消息映射机制，有关对话框和控件的知识，如何使用 Windows 图形界面绘图，了解文档/视图结构，有关数据库开发的知识及有关 Internet 的应用。

全书内容简洁明了，使用简明的语言并辅之以实例，介绍使用 Visual C++进行程序设计的各种方法和常用技巧。本书中的程序都是笔者在实践中总结提炼出来的，具有很强的实用性。

本书适合各个级别的用户使用，主要作为 Visual C++课程的培训教材，也可作为专业程序人员的参考资料，对广大学习编程的爱好者也是一本很好的学习指南。

**丛书名：流行编程软件简明教程丛书**

**书名：Visual C++简明教程**

**作者：邓力**

**出版社：海洋出版社**

**ISBN：7-5027-5324-9 / TP312**

**出版日期：2001年1月**



## 丛书前言

计算机软件技术日新月异，编程软件种类繁多且各具特色。众多的计算机爱好者都希望能成为电脑编程的行家里手，如何才能向“卡乃基殿堂”迈出正确的第一步？

为满足各级 Windows 程序开发人员、大专院校相关专业的师生和业余编程爱好者学习和使用各种流行编程软件的需求，编者在搜集了不同层次读者意见的基础上，经过仔细探讨，精心策划了本套丛书，目的在于引导读者学习二十一世纪真正的编程工具！

本丛书名为“流行编程软件简明教程丛书”，主要包括：

《C++ Builder 简明教程》

《Delphi 简明教程》

《Visual C++ 简明教程》

《Visual Basic 简明教程》

《Visual FoxPro 简明教程》

《Quick BASIC 简明教程》

《C 语言简明教程》

本套丛书针对当前最流行的开发工具，通过浅显易懂、活泼生动的语言，采用逐步引导的方式将读者带入电脑编程的殿堂，使读者轻松掌握编程的基础知识，为进一步深造打下坚实基础。生动友好的例子帮助读者了解、掌握编程技巧，同时还大大激发了读者的兴趣。

本套丛书既不是面面俱到的“功能大全”，也不是单一查询函数的“用户手册”，而是独具特色的操作和编程指导书。丛书注重对软件的主要功能和新增特性进行介绍，结合实例和项目，讲解软件的主要功能、主要原理。

我们希望，本套丛书对于读者能“抛砖引玉、触类旁通”，指引读者踏上通往编程高手之路。

总的来说，本套丛书有以下几个最优越的特点：

- ◇ “精”——精选软件，精选作者，精选图书，是值得用户收藏的精品
- ◇ “准”——紧跟软件版本更新，出版时间准，图书出版市场定位准
- ◇ “简”——教程风格简单朴素，要让读者学习起来感觉简单实用
- ◇ “明”——丛书内容十分明晰，让读者读完之后能够真正明白

我们相信：

如果你是一位初学编程的读者，本套丛书将带领你入门！

如果你是一位对编程有些了解的业余爱好者，本套丛书为你指明前进的方向！

如果你是一位软件编程专业人员或计算机行家里手，本套丛书将为你的程序锦上添花！

编者

2001年3月



# 前 言

Visual C++系列产品是开发 Windows 应用程序的有力工具，而 Visual C++ 6.0 则是 Microsoft 公司目前发布的最新版本。与 Visual C++的以前各版本相比较，Visual C++ 6.0 有重大的改进，功能更加强大，使用起来更加方便。

本书简明扼要地介绍了 Visual C++ 6.0 的使用，通过丰富的例子程序向读者介绍如何使用 MFC 开发 Windows 应用程序。全书结构安排如下：

第一章和第二章是本书的基础部分，主要介绍了 Visual C++ 6.0 的一些基本知识和如何快速地建立一个 MFC 应用程序，并介绍了 MFC 应用程序的基本运行过程。

第三章详细介绍了 MFC 应用程序框架的消息映射机制。

第四章介绍了有关对话框和控件的知识。这包括如何使用有模式对话框，如何使用各种常用的控件，最后稍稍介绍了通用对话框和无模式对话框的一些基本知识。

第五章集中介绍了有关 Windows 屏幕绘图的话题。这包括使用设备环境和 GDI 对象工作等内容，最后引导用户建立了一个屏幕保护例子程序。

考虑到读者已经有了一定的基础，第六章着重谈谈有关 Windows 设备打印方面的话题。

第七章重点介绍了 MFC 应用程序的文档/视图结构。本章的内容比较多，包括如何建立单文档类型、多文档类型和多窗口的应用程序等内容。

第八章是对第四章的必要补充，着重介绍了更多改进用户程序界面的技术，教会读者学习使用更多的控件。

第九章介绍了有关数据库的知识，这包括使用 ODBC、DAO、ADO 和 OLE DB 等技术，读者应该注意区分这些技术各自的特色。

第十章简要介绍了有关 Internet 的话题，主要是介绍如何在客户端使用 WinInet 类访问 Internet 服务器，最后还引导用户建立了一个类似于 IE5.0 的 Web 浏览器。

本书的特点在于简明扼要，原理讲解结合必要的实例，这些例子程序都是笔者根据自己学习和使用 Visual C++中的体会精心挑选的，基本上都是针对程序员在开发过程中需要最迫切、使用频率最高的内容特意定制的，可以说比较贴切地符合了初级和中级程序员的需求。另外，本书中所有例子程序的代码都经过了严格的调试和测试，读者只要跟着书中给出的步骤往下做，最终一定能够圆满地完成程序。

由于时间作者水平有限，书中肯定存在一些不足之处，恳请读者指正。

编 者  
2001 年 3 月

# 目 录

|                             |    |
|-----------------------------|----|
| 第一章 Visual C++ 6.0 概述.....  | 1  |
| 1.1 VC 6.0 的新特性.....        | 1  |
| 1.1.1 编辑器方面的新特性.....        | 1  |
| 1.1.2 编译器、连接器和调试器方面的改进..... | 1  |
| 1.1.3 MFC 类库的增强.....        | 2  |
| 1.1.4 实用工具程序.....           | 2  |
| 1.1.5 向导方面的改进.....          | 3  |
| 1.2 集成环境及基本操作.....          | 4  |
| 1.2.1 正文窗口及其操作.....         | 4  |
| 1.2.2 项目工作台窗口及其操作.....      | 5  |
| 1.2.3 输出窗口及其操作.....         | 6  |
| 1.3 菜单功能介绍.....             | 6  |
| 1.3.1 File 菜单.....          | 6  |
| 1.3.2 Edit 菜单.....          | 7  |
| 1.3.3 View 菜单.....          | 8  |
| 1.3.4 Insert 菜单.....        | 9  |
| 1.3.5 Project 菜单.....       | 9  |
| 1.3.6 Build 菜单.....         | 10 |
| 1.3.7 Tools 菜单.....         | 11 |
| 1.3.8 Window 菜单.....        | 11 |
| 1.3.9 Help 菜单.....          | 12 |
| 1.3.10 Debug 菜单.....        | 13 |
| 1.4 工具栏的使用.....             | 14 |
| 1.5 文本编辑器的使用.....           | 15 |
| 1.5.1 文件的管理.....            | 15 |
| 1.5.2 在文件中定位.....           | 17 |
| 1.5.3 对文件进行编辑.....          | 18 |
| 1.5.4 查找与替换.....            | 18 |
| 1.6 资源与标识符.....             | 20 |
| 1.6.1 资源与资源编辑器.....         | 21 |
| 1.6.2 标识符.....              | 22 |
| 1.7 本章小结.....               | 22 |

---

|                              |     |
|------------------------------|-----|
| 第二章 第一个 VC 应用程序.....         | 24  |
| 2.1 快速建立 MFC 应用 .....        | 24  |
| 2.1.1 建立新的应用程序项目 .....       | 24  |
| 2.1.2 为程序添加代码.....           | 31  |
| 2.2 分析 MFC 程序结构 .....        | 36  |
| 2.2.1 CHelloApp 应用程序类 .....  | 36  |
| 2.2.2 CMainFrame 主框架窗口类..... | 40  |
| 2.2.3 CHelloDoc 文档类.....     | 43  |
| 2.2.4 CHelloView 视图类.....    | 45  |
| 2.2.5 理解其他文件.....            | 48  |
| 2.3 MFC 程序的运行过程 .....        | 48  |
| 2.4 本章小结 .....               | 49  |
| 第三章 MFC 消息与命令 .....          | 50  |
| 3.1 消息驱动机制 .....             | 50  |
| 3.2 使用菜单工作 .....             | 51  |
| 3.3 更新菜单状态 .....             | 58  |
| 3.4 使用工具条 .....              | 63  |
| 3.5 使用加速键 .....              | 66  |
| 3.6 消息的传递 .....              | 68  |
| 3.7 本章小结 .....               | 69  |
| 第四章 对话框和控件的使用 .....          | 70  |
| 4.1 有模式对话框 .....             | 70  |
| 4.1.1 建立新项目.....             | 70  |
| 4.1.2 建立对话框资源.....           | 71  |
| 4.1.3 建立对话框类.....            | 78  |
| 4.1.4 使用对话框.....             | 83  |
| 4.1.5 与其他对象交换数据.....         | 88  |
| 4.1.6 检查和运行程序.....           | 91  |
| 4.2 无模式对话框 .....             | 94  |
| 4.2.1 建立 Exp1 项目 .....       | 94  |
| 4.2.2 修改项目资源.....            | 96  |
| 4.2.3 为对话框指定新类和加入成员函数 .....  | 97  |
| 4.2.4 代码规整.....              | 97  |
| 4.2.5 进一步理解 exp1 .....       | 105 |
| 4.3 通用对话框 .....              | 105 |
| 4.4 本章小结 .....               | 106 |

---

---

|                                   |     |
|-----------------------------------|-----|
| 第五章 图形绘制和输出 .....                 | 96  |
| 5.1 在文档窗口中绘图 .....                | 96  |
| 5.1.1 理解设备环境 .....                | 96  |
| 5.1.2 建立新的项目 .....                | 98  |
| 5.1.3 实现绘图功能 .....                | 98  |
| 5.1.4 画笔和画刷 .....                 | 104 |
| 5.1.5 实现图形拉伸 .....                | 118 |
| 5.2 在屏幕上绘图 .....                  | 122 |
| 5.2.1 屏保的概念 .....                 | 123 |
| 5.2.2 建立新的项目 .....                | 123 |
| 5.2.3 修改 InitInstance()函数 .....   | 125 |
| 5.2.4 完成设置对话框 .....               | 128 |
| 5.2.5 完成窗口类 .....                 | 132 |
| 5.3 本章小结 .....                    | 138 |
| 第六章 MFC 打印技术的应用 .....             | 139 |
| 6.1 基本打印与打印功能 .....               | 139 |
| 6.2 改变映射模式 .....                  | 142 |
| 6.3 打印多页 .....                    | 145 |
| 6.3.1 设置矩形的数目 .....               | 145 |
| 6.3.2 设置页数 .....                  | 148 |
| 6.3.3 设置每页的起点 .....               | 152 |
| 6.4 MFC 的打印进程 .....               | 153 |
| 6.5 本章小结 .....                    | 155 |
| 第七章 使用文档/视图结构 .....               | 156 |
| 7.1 分析文档/视图结构 .....               | 156 |
| 7.1.1 建立一个应用程序 .....              | 156 |
| 7.1.2 程序运行的流程 .....               | 157 |
| 7.1.3 框架窗口类 .....                 | 158 |
| 7.1.4 文档模板 .....                  | 159 |
| 7.1.5 文档类 .....                   | 160 |
| 7.1.6 视图类 .....                   | 161 |
| 7.1.7 程序员的任务 .....                | 163 |
| 7.2 单文档应用 .....                   | 163 |
| 7.2.1 单文档与多文档 .....               | 163 |
| 7.2.2 在文档中加入数据变量 .....            | 164 |
| 7.2.3 在视图中处理键盘输入 .....            | 164 |
| 7.2.4 使用视图类的 GetDocument 函数 ..... | 165 |

|                          |                                       |            |
|--------------------------|---------------------------------------|------------|
| 7.2.5                    | 将用户输入的字符存入文档.....                     | 166        |
| 7.2.6                    | 使用设备描述表显示文本.....                      | 167        |
| 7.2.7                    | 处理 WM_CREATE 消息.....                  | 168        |
| 7.2.8                    | 在屏幕上显示插入符.....                        | 169        |
| 7.2.9                    | 移动插入符.....                            | 172        |
| 7.2.10                   | 用 DeleteContents 函数进行数据清除.....        | 173        |
| 7.2.11                   | 用 OnNewDocument 函数进行初始化.....          | 174        |
| 7.2.12                   | 用鼠标定位插入符.....                         | 175        |
| 7.3                      | 多文档应用.....                            | 178        |
| 7.3.1                    | 建立一个多文档的应用.....                       | 178        |
| 7.3.2                    | 分析 AppWizard 产生的 MDI 框架程序.....        | 179        |
| 7.3.3                    | 增强文本编辑器的功能.....                       | 182        |
| 7.3.4                    | 设置文档的修改标志.....                        | 183        |
| 7.3.5                    | 修改视图类的 OnDraw 函数.....                 | 185        |
| 7.4                      | 多窗口应用.....                            | 186        |
| 7.4.1                    | 程序框架实现的功能.....                        | 186        |
| 7.4.2                    | 使文档和视图保持一致.....                       | 187        |
| 7.4.3                    | 在 OnChar 函数中加入 UpdateAllViews 函数..... | 188        |
| 7.4.4                    | 修改视图类的 OnUpdate 成员函数.....             | 189        |
| 7.4.5                    | 视图类的 OnInitialUpdate 函数.....          | 190        |
| 7.5                      | 本章小结.....                             | 191        |
| <b>第八章 改进应用程序界面.....</b> |                                       | <b>192</b> |
| 8.1                      | 控制条类.....                             | 192        |
| 8.1.1                    | 控制条.....                              | 192        |
| 8.1.2                    | 工具条.....                              | 193        |
| 8.1.3                    | 状态条.....                              | 193        |
| 8.1.4                    | 对话框条.....                             | 193        |
| 8.1.5                    | 集合条.....                              | 194        |
| 8.2                      | 工具条和状态条.....                          | 194        |
| 8.2.1                    | 缺省的工具条与状态条.....                       | 194        |
| 8.2.2                    | 创建自己的工具条.....                         | 199        |
| 8.2.3                    | 向状态条中添加指示器.....                       | 208        |
| 8.3                      | 对话框条.....                             | 213        |
| 8.4                      | 集合条.....                              | 215        |
| 8.4.1                    | 建立 AdvBar 程序框架.....                   | 215        |
| 8.4.2                    | 建立新的工具条.....                          | 216        |
| 8.4.3                    | 建立集合条.....                            | 221        |
| 8.5                      | 本章小结.....                             | 225        |

---

---

|                                 |     |
|---------------------------------|-----|
| 第九章 数据库的开发和使用 .....             | 226 |
| 9.1 关系数据库模型 .....               | 226 |
| 9.1.1 数据结构 .....                | 226 |
| 9.1.2 完整性规则 .....               | 227 |
| 9.1.3 数据操作 .....                | 227 |
| 9.1.4 SQL 语言 .....              | 228 |
| 9.2 使用 ODBC .....               | 231 |
| 9.2.1 ODBC 概述 .....             | 231 |
| 9.2.2 创建 ODBC 数据库应用程序 .....     | 234 |
| 9.2.3 为数据库应用程序创建视图 .....        | 238 |
| 9.2.4 应用程序是如何工作的 .....          | 240 |
| 9.2.5 遍历、添加、修改和删除记录 .....       | 243 |
| 9.2.6 统计函数和多表连接 .....           | 253 |
| 9.2.7 直接使用 SQL 语句 .....         | 259 |
| 9.2.8 使用 CDatabase 进行事务处理 ..... | 260 |
| 9.3 使用 DAO .....                | 261 |
| 9.3.1 DAO 概述 .....              | 261 |
| 9.3.2 MFC DAO 类 .....           | 261 |
| 9.3.3 创建 DAO 数据库应用程序 .....      | 262 |
| 9.3.4 理解派生的记录集类 .....           | 265 |
| 9.4 其它数据库技术简介 .....             | 268 |
| 9.4.1 OLE DB 技术 .....           | 268 |
| 9.4.2 ADO 技术 .....              | 269 |
| 9.5 本章小结 .....                  | 269 |
| 第十章 Internet 编程 .....           | 270 |
| 10.1 使用 WinInet 类 .....         | 270 |
| 10.1.1 建立应用程序框架 .....           | 271 |
| 10.1.2 与 Internet 连接 .....      | 273 |
| 10.2 制作 Web 浏览器 .....           | 278 |
| 10.2.1 建立应用程序框架 .....           | 278 |
| 10.2.2 浏览 Web 页 .....           | 279 |
| 10.2.3 改善程序的界面 .....            | 285 |
| 10.2.4 检查应用程序功能 .....           | 292 |
| 10.3 本章小结 .....                 | 293 |

# 第一章 Visual C++ 6.0 概述

“真正的程序员用 Visual C++!”

Visual C++是当今最被广泛使用的可视化编程环境，为我们提供了一种方便、快捷的 Windows 应用程序开发工具。它使用了 Microsoft Windows 图形用户界面的许多先进特性和设计思想，采用了弹性可重复利用的完整的面向对象程序语言（Object-Oriented Language）、当今世界上最快的编辑器、最为领先的数据库技术。

Visual C++ 6.0 是 Visual C++系列的最新版本，是为 Windows 98 或 Windows 2000 等 32 位操作系统开发应用程序用的编程工具，功能强大，界面友好，操作方便。本章为全书的第一章，主要是对 Visual C++ 6.0 做一些简要的说明，介绍 Visual C++ 6.0 的基本特性。读者在阅读完本章后，将对 Visual C++ 6.0 有个大概的了解。

在本书以下的叙述中，为了简单起见，将 Visual C++ 6.0 简称为 VC6.0。

## 1.1 VC 6.0 的新特性

与 Visual C++系列以前的版本相比而言，VC6.0 提供了许多新的特性，大致可以分为以下几方面。

### 1.1.1 编辑器方面的新特性

VC6.0 在集成开发环境的编辑器做了一些改进，以方便用户快速准确地编辑代码和资源，这些新特性主要包括：

#### ■ 自动完成语句功能

在用户编辑代码时，编辑器根据光标当前位置判断作用的类或对象，在一个下拉列表中显示相应的内容，如类的成员、函数原型、标识符定义等等。自动完成语句功能减轻了用户在输入长的类名或成员对象名时的繁琐工作，方便了用户的使用。

#### ■ 快速宏录制

用户可以将集成开发环境中特定的连续操作定义为宏，在需要再次使用类似操作的时候只需调用已录制好的宏即可。

#### ■ 支持 IE5 新控件的资源编辑器

新版本的资源编辑器支持 IE5 提供的四种新控件，在资源编辑器中可以方便地将这些新控件添加到工具栏或者对话框中。

### 1.1.2 编译器、连接器和调试器方面的改进

VC6.0 在集成开发环境的编译器、连接器和调试器方面也做了大的改进，其目的就是使得用户的应用程序运行起来更快、更稳定，调试起来更方便。

- 编译器方面的改进

包括新添加的“\_assume”关键字、增强对内联函数的控制、新增并更新了警告、加强了运行时刻的错误检测等内容。

- 连接器方面的改进

包括延迟加载外部支持、增添了新的连接选项和修正了外部函数的接口以减小文件尺寸等内容。

- 调试器方面的改进

包括改善了反汇编输出、改善了指针对象的显示、支持进程中的远程过程调用等内容。

### 1.1.3 MFC 类库的增强

VC6.0 在 MFC 类库方面做了许多改进，这包括一系列新的或改进过的类、对新技术的支持等等。这些改进大大增强了 MFC 应用程序的功能，主要包括以下方面的内容：

- Active 文档容器

与 OLE 容器相区别，Active 文档容器能够一次激活整个文档，文档占据了容器应用程序的整个框架窗口。通过在 AppWizard 中选择合适的选项，很容易就能够使得用户开发的应用程序成为 Active 文档容器。

- 动态 HTML 控件和 CHtmlView 类

动态 HTML 控件就是指 WebBrowser 控件，提供了浏览 HTML 页面的能力。CHtmlView 类封装了该控件，并能够在应用程序的视图中显示 HTML 页面。

- 支持 IE5 提供的新控件

IE5 提供了四种类型的新控件，MFC 为这些新类型的控件提供了类的支持，这包括 CComboBoxEx 类、CDateTimeCtrl 类、CIPAddressCtrl 类和 CMonthCalCtrl 等类，分别支持对应类型的控件。

- 新提供的类和改进的类

MFC 提供了许多新的类，除了上述四个类之外，还包括 COleDBRecordView 类、COleDocObjectItem 类、CPropertySheetEx 类、CPropertyPageEx 类、CReBar 类和 CReBarCtrl 类等；改进过的类就更多了，典型的有 CString 类、CMenu 类、CToolBar 类和 CListCtrl 类等等。

- 新的全局函数

MFC 还提供了新的全局函数 AfxCheckError()和 AfxDumpStack()，这都是为用户调试程序服务的。用户可以在程序的任何位置调用这些全局函数。

- 性能调整

静态链接的 MFC 库的尺寸减小了。

### 1.1.4 实用工具程序

VC6.0 中提供了一些新的实用工具程序，同时也对原有的一部分工具程序做了改进。

- 组件管理器

组件管理器帮助用户管理和共享软件工具或者可重用组件，其目的就是为了更好地支持代码或资源的重用与管理、协调开发工具的使用和管理软件小组的共同开发活动等等。

### ■ HTML 帮助

HTML 帮助是一种新格式的帮助系统，用户可以在 Web 页中集成基于 HTML 格式的上下文相关帮助。HTML 帮助系统比传统 Windows 帮助系统提供了更多的功能，VC6.0 附带的 MSDN 就是 HTML 帮助系统的典范。

### ■ 安装程序开发工具

为交流而开发的软件必须提供一整套的安装程序，InstallShield 就是用户书写安装程序的好帮手。随 VC6.0 提供的 InstallShield 6.0 是该工具程序的最新版本，也是为 VC6.0 度身定制的。

### ■ 控件测试容器

新版本的控件测试容器更加稳定，并且支持 OCX96 新特性。控件测试容器还允许用户使用 VBScript 语言书写测试脚本程序。

## 1.1.5 向导方面的改进

在使用 Visual C++ 系列开发应用程序的过程中，用户在很多情况下都需要向导的帮助，VC6.0 在向导方面做了许多改进。

### ■ 改进的 AppWizard

改进的 AppWizard 为用户定制 MFC 应用程序提供了更多的选择。新的 AppWizard 支持不带文档/视图结构的应用程序，可以为应用程序提供 Active 文档容器或者服务器功能，支持 ReBar 控件和网络浏览器风格的应用程序等。

### ■ 支持 OLE DB

在 AppWizard 中选择数据库支持时，除了选择 DAO 和 ODBC 外，现在还可以选择 OLE DB，这可通过使用 MFC 提供的 OLE DB 模板实现。

### ■ 为帮助文件定制编译规则

在建立应用程序的帮助文件时，AppWizard 用定制编译规则的方法取代了原来使用的 MakeHelp.bat，这使得帮助文件的编译只有在需要时才会进行。

### ■ 删除成员函数

在 VC6.0 中，用户可以通过 WizardBar 或者 ClassView 删除类的不再需要的成员函数，这虽然只是一个很小的功能，但是很实用。

### ■ 新的项目类型

定制的 AppWizard 现在可以支持静态链接库、控制台应用程序和 ActiveX 控件等多种新的项目类型。

### ■ 为非 MFC 程序提供的向导

VC6.0 为许多非 MFC 程序项目也提供了向导支持，这包括为 Win32 应用程序、动态链接库、控制台程序提供的向导等等。

### 提示：

本节中只是简单地介绍了 VC6.0 的一些新的特性，有关新特性的更多信息，请参阅 VC6.0 联机手册中的“*What's new*”部分的内容。

## 1.2 集成环境及基本操作

Microsoft Develop Studio 集成开发环境功能强大，但同时其窗口也比较复杂，包含有多个子窗口和工具栏，而且子窗口又包含了若干标签和选项卡，有必要对集成开发环境的基本使用做一些介绍。

集成开发环境的基本外观如图 1-1 所示。

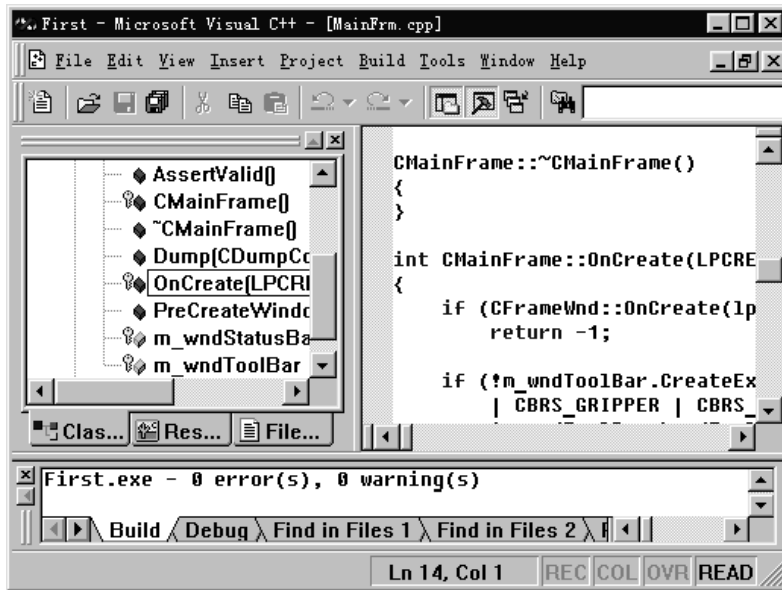


图 1-1 集成开发环境的基本外观

### 1.2.1 正文窗口及其操作

各种程序文件、资源文件和文档文件都可以在正文窗口中打开并相应地进行编辑。用户可以同时打开多个正文窗口（这就是所谓的多文档界面），每个窗口都可以通过其系统菜单或右上角的按钮进行最大化、最小化、复原、关闭、移动和改变尺寸等操作，并且可以通过在 Window 菜单中选定特定窗口或通过鼠标激活到想切换的窗口。

同其他 Windows 应用程序类似，Window 菜单下的 New window 命令将打开当前文档的另一窗口；Split 命令将当前窗口分成四个小窗口；Close 和 Close all 命令将关闭正文窗口。Next 命令将切换到下一个正文窗口；Previous 命令将切换到上一个正文窗口。Cascade 命令将层叠排列各正文窗口；Tile horizontally 和 Tile vertically 命令将水平平铺或垂直平铺各正文窗口。

也可以通过选择 Window 菜单下的 Windows 命令弹出对话框来操纵正文窗口，如图 1-2 所示。

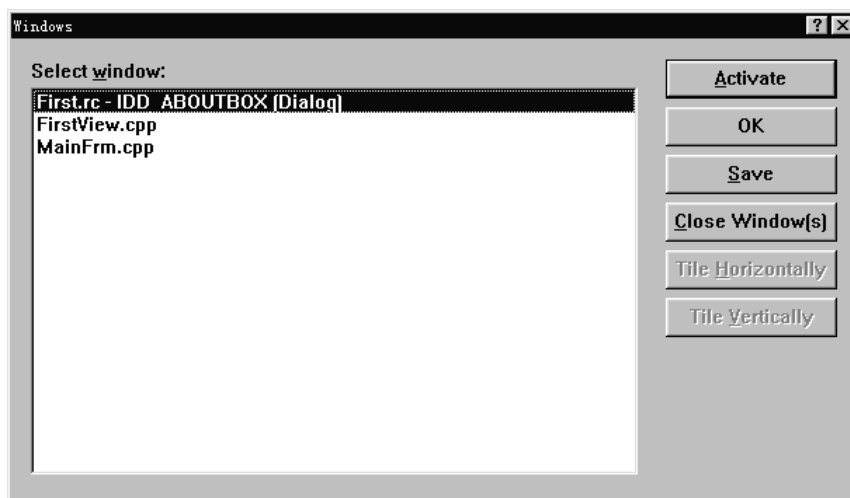


图 1-2 Windows 对话框

在 Windows 对话框中，左边的列表框列出了当前所有打开的正文窗口。选中一个正文窗口，单击 **Activate** 按钮将使得选中的正文窗口成为当前窗口；单击 **OK** 按钮将退出本对话框；单击 **Save** 按钮将储存选中的正文窗口的内容；单击 **Close Window** 将关闭该正文窗口。如果选中了多个正文窗口，则 **Tile Horizontally** 和 **Tile Vertically** 按钮变为有效，单击可以水平平铺或垂直平铺选中的窗口。

这个对话框看起来与菜单命令功能重复，在当前打开的正文窗口少的时候确实如此。然而如果打开的窗口比较多的话，比如说超过 10 个，这时 Windows 对话框就显得比较方便了，不过同时打开超过 10 个窗口的机会似乎并不多。

### 1.2.2 项目工作台窗口及其操作

项目工作台窗口在正文窗口的左边。在项目工作台窗口中，一般有三个标签：**Classview**（类视图）、**Resourceview**（资源视图）和 **Fileview**（文件视图）。单击某个标签就会显示相应的视图。

在 **Classview** 中，显示了当前工作空间（**Workspace**）的项目，以及项目中的类和类的成员变量、成员函数。就象在资源管理器中一样，**Classview** 中的对象可以一层一层地展开，直至最底层的对象。如果在 **Classview** 中双击类名，将在正文窗口中打开定义了该类的头文件；如果双击变量名，将在正文窗口中打开定义该变量的头文件，并高亮显示在正文窗口的中间；如果双击函数名，则将在正文窗口中打开该函数的声明，并高亮显示。利用这一特点，可以快速地浏览类的组成并跳到合适的地方。另外，用鼠标右键单击某个对象，将弹出一个快捷菜单，显示了该对象可以执行的命令，这也是很方便的。

**Resourceview** 显示了当前项目所用到的资源状况，如菜单、工具栏、对话框等等。在 **Resourceview** 中，同样可以展开，双击某个对象将在正文窗口打开该对象。

**Fileview** 几乎就是一个缩微版的资源管理器，不过它只显示当前项目所用到的文件。利用它可以快速打开指定文件。

**注意：**

与 Visual C++ 5.0 不同，这里并没有 InfoViewer，Microsoft 把所有的帮助信息都集成到了另外一个可单独运行的程序——MSDN 中。如果用户在安装 Visual C++ 6.0 的时候同时安装了 MSDN，那么用户应该到 MSDN 中查询帮助信息。

1.2.3 输出窗口及其操作

输出窗口在集成开发环境的底部，包括了 Build（编译）、Debug（调试）、Find in files 1（文件查询结果 1）、Find in files 2（文件查询结果 2）、Results（结果）和 SQL Debugging（SQL 调试）等标签。Build 标签中显示了编译和连接的进度和错误信息。Debug 标签中显示了调试信息。输出窗口的操作方式类似于项目工作台窗口。

## 1.3 菜单功能介绍

集成开发环境的菜单提供各种各样的命令，在开发应用程序的过程中将大量地使用菜单命令。因此，先简单地介绍一下各个菜单的基本功能是很有必要的。下面将逐一介绍集成开发环境的菜单。

1.3.1 File 菜单

File 菜单中的命令主要执行新建、打开或关闭文件或工作空间，另外还有打印和打印设置，最近使用过的文件和工作空间列表等。如图 1-3 所示。

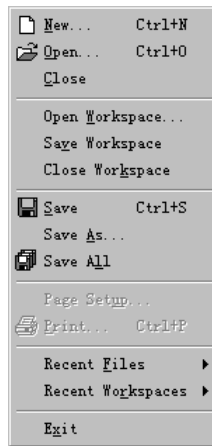


图 1-3 File 菜单

File 菜单命令的快捷键和功能的简要介绍如表 1-1 所示。

表 1-1 File 菜单命令的快捷键和功能说明

| 菜单命令 | 快捷键    | 功能说明              |
|------|--------|-------------------|
| New  | Ctrl+N | 新建文件、项目、工作空间或其他文档 |
| Open | Ctrl+O | 打开一个已存在的文件        |

续 表

| 菜单命令             | 快捷键    | 功能说明         |
|------------------|--------|--------------|
| Close            | ——     | 关闭当前文件       |
| Open Workspace   | ——     | 打开已存在的工作空间   |
| Save Workspace   | ——     | 储存当前工作空间     |
| Close Workspace  | ——     | 关闭当前工作空间     |
| Save             | Ctrl+S | 储存当前文件       |
| Save As          | ——     | 用新的文件名储存当前文件 |
| Save All         | ——     | 储存所有文件       |
| Page Setup       | ——     | 设置文件的打印页面    |
| Print            | Ctrl+P | 打印文件的全部或部分   |
| Recent Files     | ——     | 最近使用过的文件列表   |
| Recent Workspace | ——     | 最近使用过的工作空间列表 |
| Exit             | ——     | 退出集成开发环境     |

### 1.3.2 Edit 菜单

Edit 菜单中的命令主要有撤消、恢复上一步的操作，剪切、复制、粘贴和删除选定区域，查找与替换字符串，定位与标签，断点设置等普通命令。

Microsoft Visual Studio 6.0 版还增添了一些新的功能，如最下方的四个命令。用户如果使用过 Visual C++ 的以前的版本，是否在输入一个名称较长或拥有好几个参数的函数时遇到过困难，经常要在帮助文档和编辑窗口间进行切换。现在好了，在用户进行输入的时候，Visual Studio 可以弹出一个关于当前对象的说明，如对象的各个成员、函数原型、参数类型等等，甚至可以帮助你完成要输入的字符。看来在 Visual Studio 6.0 中，Tooltips 的功能是大大地增强了。Edit 菜单如图 1-4 所示：

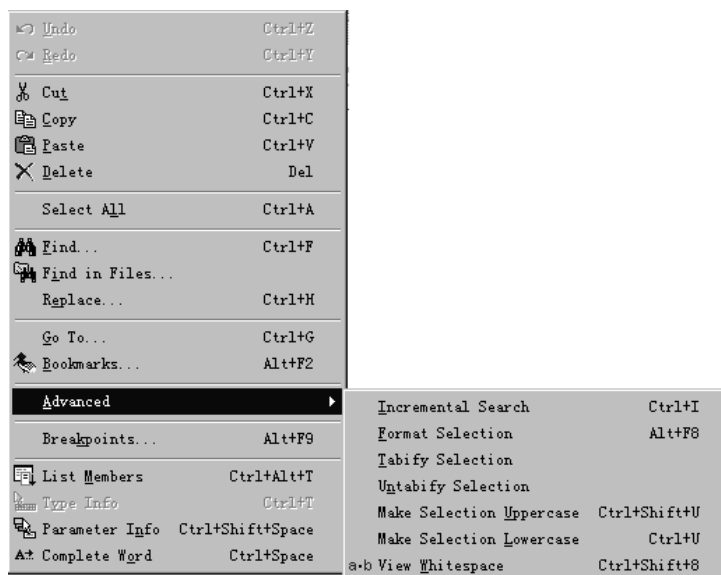


图 1-4 Edit 菜单

各命令的快捷键和功能说明如表 1-2 所示。

表 1-2 Edit 菜单命令的快捷键和功能说明

| 菜单命令                              | 快捷键              | 功能说明                     |
|-----------------------------------|------------------|--------------------------|
| Undo                              | Ctrl+Z           | 撤消上一次操作                  |
| Redo                              | Ctrl+Y           | 恢复上一次撤消的操作               |
| Cut                               | Ctrl+X           | 将选中的内容剪切至剪贴板             |
| Copy                              | Ctrl+C           | 将选中的内容复制至剪贴板             |
| Paste                             | Ctrl+V           | 从剪贴板中粘贴内容至光标处            |
| Delete                            | Del              | 删除选中的内容                  |
| Select All                        | Ctrl+A           | 选中所有的内容                  |
| Find                              | Ctrl+F           | 在当前打开的正文窗口中查找字符串         |
| Find in files                     | ——               | 在当前项目的具有指定扩展名的文件中查找字符串   |
| Replace                           | Ctrl+H           | 查找指定字符串并用新的字符串代替         |
| Go To                             | Ctrl+G           | 跳转到指定的行、标签或地址等           |
| Bookmarks                         | Alt+F2           | 将当前位置制作标签                |
| Advanced\Incremental Search       | Ctrl+I           | 直接对键盘输入的字符串进行查找          |
| Advanced\Format Selection         | Alt+F8           | 按格式进行选择                  |
| Advanced\Tabify Selection         | ——               | 对选中内容按制表位置定位             |
| Advanced\Untabify Selection       | ——               | 取消定位                     |
| Advanced\Make Selection Uppercase | Ctrl+Shift+U     | 将选中的所有字符转换为大写            |
| Advanced\Make Selection Lowercase | Ctrl+U           | 将选中的所有字符转换为小写            |
| Advanced\View Whitespace          | Ctrl+Shift+8     | 显示空格、制表位等空白字符            |
| Breakpoints                       | Alt+F9           | 设置或清除程序的断点               |
| List Members                      | Ctrl+Alt+T       | 列出光标左边的类或结构的有效的成员变量和成员函数 |
| Type Info                         | Ctrl+T           | 显示标识符的完整的声明              |
| Parameter Info                    | Ctrl+Shift+Space | 显示光标所在处函数的完整声明，包括参数      |
| Complete Word                     | Ctrl+Space       | 由编辑器自动完成待输入的单词           |

### 1.3.3 View 菜单

View 菜单如图 1-5 所示。

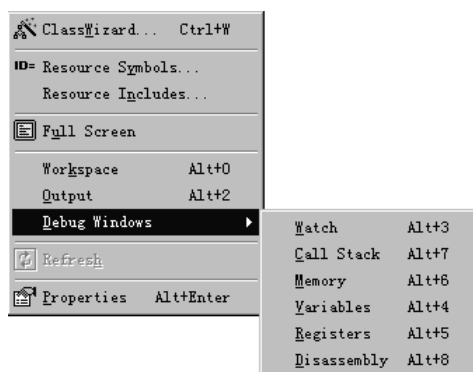


图 1-5 View 菜单

View 菜单中的命令主要功能是显示/隐藏指定的子窗口，全屏编辑模式切换等等。各命令的快捷键和功能说明如表 1-3 所示。