

精彩 C++ Builder 6 程序设计

吴逸贤 吴目诚 编著

北 京

图字：01-2002-0926

内 容 简 介

C++ Builder 是最知名的可视化 C++ 应用程序开发工具之一，它具有易学易用的操作接口，功能强大的应用组件，它将 C 语言简洁、快速的优点与面向对象的软件工程设计方法相结合，使程序设计师可以最少的程序代码来建立高效率的 Windows 应用程序。

本书介绍 Borland 公司 2002 年推出的最新版本 C++ Builder 6，内容涵盖了 C++ Builder 的基本知识和操作方法、程序设计的基本概念、程序代码的编写技巧、鼠标和键盘等事件程序的控制技巧、C++ Builder 的高级功能、开发应用程序经常需要的技巧，以及 15 个有一定难度的应用专题，通过这些最适合学习的应用程序范例，可使用户练就扎实的编程技术。

本书经过精心策划，采取循序渐进的方式，并充分应用练习范例，引导用户迅速掌握 C++ 的程序语法与技巧。本书适合初学者和有一定编程经验的读者使用。

本书繁体字版名为《精彩 C++ Builder 6 程式设计》，由知城数位科技股份有限公司出版。本书中文简体字版由知城数位科技股份有限公司授权科学出版社独家出版。未经本书原出版者和本书出版者书面许可，任何单位和个人均不得以任何形式或任何手段复制或传播本书的部分或全部内容。

图书在版编目(CIP)数据

精彩 C++ Builder 6 程序设计/吴逸贤编著.—北京：科学出版社，2003.6

ISBN 7-03-011608-9

I. 精... II. 吴... III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(2003)第 044646 号

责任编辑：李佩乾/责任校对：都 岚
责任印制：吕春珉/封面制作：一克米工作室

出版

北京东黄城根北街 16 号
邮政编码：100717

印刷

科学出版社发行 各地新华书店经销

*

2003 年 6 月第 一 版 开本：787×1092 1/16
2003 年 6 月第一次印刷 印张：33
印数：1—4000 字数：792 000

定价：49.00 元

序

C++ Builder 作为软件业界重要的开发工具，是最成熟的图形界面 C++ 程序开发环境，也是最有效率的 C++ 程序开发工具。想要成为程序设计师，就有必要学习使用 C++ Builder 开发应用程序。

这是一本 C++ 程序设计宝典，如果您具有程序设计的经验，本书将引导您迅速掌握 C++ 的程序语法与技巧！您也可以完全没有编程经验，从本书介绍的 C++ 程序中，一步一步练就程序设计的基本功！

本书经过精心的分析与规划，由浅入深、循序渐进，充分应用练习范例，展现 C++ Builder 6 的各种功能，全书共 19 章，分为基本操作、程序设计基础、应用与进阶和专题设计等 4 篇，各篇的内容如下：

- 基本操作篇：旨在帮助读者学习 C++ Builder 的基本知识和操作方法，内容包括窗体设计、组件制作，以及程序调试与打包等功能。
- 程序设计基础篇：介绍程序语言的基本概念，以 C++ 为例，说明程序设计的基本概念，并以实际的程序代码介绍各种数据表示法、控制结构、数组、排序、指针与内存配置、对象与类、建立对象、自定义对象等程序代码的编写技巧，以及鼠标、键盘等事件程序的控制技巧。
- 应用与进阶篇：介绍 C++ Builder 的高级功能，以及开发应用程序经常需要的技巧，包括绘图、文件系统与数据库、数据打印、网络程序设计、菜单、MDI 以及 Windows API 的调用技巧等。
- 专题设计篇：提供了 15 个有一定难度的应用专题，它们是最适合学习的程序应用专题，可以帮助您练就扎实的编程技术。本篇包括智力游戏专题和高互动型游戏专题等两章，内容包括拼图游戏、猜猜看、配对游戏、配对记忆游戏、搬搬看、角子老虎、智能盘游戏、英文单字练习、踩地雷游戏、神枪手、猴仔兵团、钓鱼高手、打砖块游戏、青蛙过街、键盘操作练习等，值得您细细品味，从中领略程序设计的艺术。

我们期待通过这本书和您做个朋友，让这本书陪伴您进入 C++ Builder 程序设计的缤纷世界。读完这本书，相信您一定可以体会到我们的用心与专业。

吴逸贤 吴目诚

HaBook 网奕数字科技

habook@habook.com.tw

目 录

第 I 部分 基本操作篇

第 1 章 C++ Builder 基本操作.....1	
1.1 认识 C++ Builder 6.....1	
1.1.1 认识 C++ Builder.....1	
1.1.2 认识 C++ Builder 的窗口.....3	
1.1.3 认识工具按钮.....6	
1.2 建立一个项目.....7	
1.2.1 建立新项目.....7	
1.2.2 打开项目文件.....9	
1.2.3 新建与删除窗体.....10	
1.2.4 在窗体上新建组件.....12	
1.3 编写与测试程序.....13	
1.3.1 在程序代码窗口输入程序代码.....13	
1.3.2 使用按钮来控制程序的执行.....16	
1.3.3 输入组件与计算结果.....17	
1.3.4 程序代码窗口的辅助编辑功能.....19	
1.4 问题与解答.....21	
第 2 章 窗体与组件设计.....22	
2.1 窗体编辑技巧.....22	
2.1.1 设置窗体的尺寸.....22	
2.1.2 设置窗体的背景图案.....23	
2.2 组件工具箱.....25	
2.2.1 认识组件工具箱.....25	
2.2.2 新建组件工具按钮.....35	
2.3 在窗体上建立组件.....36	
2.3.1 建立与设置文字标签.....36	
2.3.2 建立与设置命令按钮.....38	
2.3.3 建立基本图案.....41	
2.3.4 建立与设置文本框.....43	
2.3.5 建立滚动条组件.....45	
2.3.6 建立与应用列表框.....47	
2.3.7 建立与应用下拉列表框.....51	
2.3.8 框架、单选按钮与复选框.....53	

2.4 组件的编排.....57	
2.4.1 复制组件.....57	
2.4.2 对齐组件位置.....58	
2.4.3 调整间距.....59	
2.4.4 调整图层.....60	
2.5 问题与解答.....61	

第 3 章 程序调试与打包.....63

3.1 C++ Builder 的调试工具.....63	
3.1.1 利用 ShowMessage 查看变量.....63	
3.1.2 局部变量窗口.....64	
3.1.3 监看窗口.....65	
3.1.4 查看程序的相关模块.....68	
3.2 作品输出与打包.....70	
3.2.1 生成可执行文件.....70	
3.2.2 安装封装程序.....71	
3.2.3 使用封装程序.....76	
3.2.4 安装与执行应用程序.....81	
3.3 问题与解答.....84	

第 II 部分 程序设计基础篇

第 4 章 C++与程序基本概念.....85

4.1 程序语言的分级.....85	
4.2 高级程序语言.....87	
4.3 面向过程与面向对象.....93	
4.4 流程图.....94	
4.4.1 认识流程图符号.....94	
4.4.2 程序与流程图.....96	
4.5 逻辑控制结构.....97	
4.5.1 顺序.....97	
4.5.2 选择.....97	
4.5.3 循环.....99	
4.6 子程序与函数.....100	
4.6.1 子程序.....100	

4.6.2 参数传递	101	7.1.4 参数传递与返回值	130
4.6.3 函数	101	7.2 值调用与地址调用	132
4.7 问题与解答	102	7.2.1 值调用	132
第 5 章 数据表示法与表达式	104	7.2.2 地址调用	132
5.1 数据类型与变量	104	7.3 内置函数	134
5.1.1 数据类型	104	7.3.1 数学函数	134
5.1.2 变量	105	7.3.2 字符与字符串函数	135
5.1.3 声明新的变量名称	105	7.3.3 日期时间函数	135
5.1.4 变量数据类型转换	106	7.4 递归程序	136
5.1.5 变量的使用范围	106	7.5 问题与解答	137
5.1.6 变量的作用域	108	第 8 章 数组、字符串与排序	139
5.2 运算符与语法	109	8.1 设置数组数据	139
5.2.1 运算符	109	8.1.1 认识数组	139
5.2.2 运算符的优先次序	111	8.1.2 声明一维数组	140
5.2.3 运算符练习	112	8.1.3 声明二维以上的数组	141
5.3 自定义数据类型	113	8.1.4 把数据放入二维数组中	141
5.3.1 定义数据类型	113	8.2 字符与字符串数组	143
5.3.2 声明自定义数据类型变量	114	8.2.1 字符数组的声明与操作	143
5.4 组件属性与组件变量	115	8.2.2 二维字符数组	144
5.4.1 使用窗口设置组件属性	115	8.2.3 不指定大小的字符数组	144
5.4.2 使用程序代码设置组件属性	116	8.2.4 AnsiString 字符数组	144
5.5 问题与解答	116	8.3 自定义数据类型数组	145
第 6 章 控制结构	118	8.3.1 结构数组	145
6.1 选择结构	118	8.3.2 结构中的数组元素	146
6.1.1 单向选择结构	118	8.4 随机、排序与查找	146
6.1.2 双向选择结构	120	8.4.1 随机	146
6.1.3 多向选择结构	121	8.4.2 排序	147
6.2 循环结构	123	8.4.3 查找	148
6.2.1 记数循环	123	8.5 问题与解答	149
6.2.2 条件循环	124	第 9 章 指针与内存配置	151
6.2.3 多重循环	124	9.1 指针的声明与使用	151
6.3 问题与解答	125	9.1.1 认识指针	151
第 7 章 子程序和内置函数	127	9.1.2 使用指针来进行运算	153
7.1 函数与子程序	127	9.1.3 双重指针	154
7.1.1 子程序的格式与类型	127	9.2 数组与指针	154
7.1.2 建立子程序	128	9.2.1 字符数组与指针	154
7.1.3 调用子程序	129	9.2.2 数值数组与指针	156
		9.2.3 指针数组	157

9.3 参数传递与指针函数.....	157	第 11 章 对象与事件控制技巧.....	193
9.3.1 传递指针参数.....	157	11.1 组件控制技巧.....	193
9.3.2 指针函数.....	158	11.1.1 利用时间变化来移动组件.....	193
9.4 转换数据类型.....	159	11.1.2 同时控制多个不同移动速度的组件.....	194
9.4.1 整数和字符类型转换.....	159	11.1.3 应用类随机产生不断上移的组件.....	197
9.4.2 用指针解析整数的表示法.....	160	11.1.4 根据指定路径自由移动的组件.....	200
9.5 内存分配.....	161	11.1.5 应用三角函数制作圆形运动.....	202
9.5.1 动态分配整数数组.....	161	11.1.6 随机显示图片.....	205
9.5.2 内存分配命令 malloc().....	161	11.1.7 画面上弹跳的小球.....	206
9.6 问题与解答.....	162	11.2 事件程序.....	208
第 10 章 对象与类.....	164	11.2.1 认识事件.....	208
10.1 C++ Builder 的对象.....	164	11.2.2 常用事件名称与功能.....	209
10.1.1 认识对象.....	164	11.3 鼠标控制技巧.....	233
10.1.2 组件的从属关系.....	165	11.3.1 测试鼠标按键的状态.....	223
10.1.3 调整组件的层级.....	166	11.3.2 显示鼠标坐标位置.....	224
10.2 在执行阶段产生组件.....	169	11.3.3 用鼠标指针拉动布景.....	225
10.2.1 在执行阶段产生新组件.....	169	11.3.4 简易打地鼠程序.....	226
10.2.2 在执行阶段复制组件.....	171	11.3.5 拖动组件与配对.....	229
10.3 认识类.....	172	11.3.6 坐标位置记录器.....	232
10.3.1 类与对象的关系.....	172	11.4 键盘控制技巧.....	233
10.3.2 定义一个简单的类.....	173	11.4.1 键盘按键的键值.....	234
10.3.3 包含属性和方法的类.....	174	11.4.2 使用 Enter 键输入答案.....	235
10.3.4 构造函数与析构函数.....	175	11.4.3 利用键盘按键拉动布景.....	237
10.3.5 使用方法返回变量值.....	177	11.4.4 仿真键盘按键状态.....	239
10.3.6 函数成员的重载与参数传递.....	177	11.5 问题与解答.....	241
10.3.7 类中的静态变量.....	178		
10.3.8 类声明外的函数成员.....	179		
10.3.9 设置初值的精简写法.....	179		
10.4 面向对象程序设计.....	180		
10.4.1 面向对象程序语言的基本特性.....	180		
10.4.2 封装.....	180		
10.4.3 继承.....	181		
10.4.4 修改继承的成员特性.....	184		
10.4.5 多重继承.....	185		
10.4.6 虚拟继承.....	186		
10.4.7 多态与虚拟.....	188		
10.5 问题与解答.....	190		
		第 III 部分 应用与进阶篇	
		第 12 章 绘图指令的认识与应用.....	243
		12.1 绘图指令.....	243
		12.1.1 设置颜色.....	243
		12.1.2 基本绘图方法.....	244
		12.1.3 设置图形的属性.....	247
		12.2 交互式绘图技巧.....	249
		12.2.1 涂鸦板.....	249
		12.2.2 画圆形互动程序.....	252

12.2.3	画正三角形互动程序.....	254	14.1	基本打印技巧.....	317
12.2.4	画正多角形互动程序.....	256	14.1.1	打印机的设置.....	317
12.3	动画技巧.....	259	14.1.2	打印属性设置.....	318
12.3.1	小向导.....	259	14.2	输出图形到 Printer 控件.....	322
12.3.2	弹跳中的小圆球.....	261	14.2.1	在 Printer 控件上 使用绘图指令.....	322
12.4	图形的存储与输出.....	263	14.2.2	打印图形文件.....	323
12.4.1	在 Image 控件上 使用绘图指令.....	263	14.3	问题与解答.....	325
12.4.2	存储图形.....	265	第 15 章	菜单与 MDI 设计.....	326
12.4.3	打印绘图.....	266	15.1	菜单的建立与编辑.....	326
12.5	图像与图像串行.....	267	15.1.1	认识菜单编辑器.....	326
12.5.1	图像复制与分割.....	267	15.1.2	菜单项目与程序代码结合.....	328
12.5.2	建立图像串行 TImageList.....	269	15.1.3	在菜单项目中打勾.....	330
12.6	音效与影片.....	271	15.1.4	淡化处理菜单项目.....	331
12.6.1	播放 AVI 影片.....	272	15.2	菜单的高级应用.....	332
12.6.2	播放 WAV 音效.....	275	15.2.1	快捷菜单.....	332
12.6.3	使用 OLE 控件播放音效.....	277	15.2.2	建立多层式菜单.....	334
12.7	问题与解答.....	280	15.2.3	在菜单选项上显示图像.....	336
第 13 章	文件存取与数据库.....	281	15.2.4	动态显示菜单项目.....	338
13.1	顺序文件与随机文件.....	281	15.3	MDI 多文档界面设计.....	340
13.1.1	读取顺序文件.....	281	15.3.1	建立 MDI 窗体.....	340
13.1.2	把数据输出成为顺序文件.....	284	15.3.2	用菜单切换子窗体.....	341
13.1.3	建立随机文件.....	285	15.3.3	用菜单排列子窗体.....	343
13.1.4	读取随机文件.....	287	15.3.4	在执行阶段建立子窗体.....	344
13.2	文本文件的存取.....	289	15.3.5	使用向导建立 MDI 应用程序.....	346
13.3	数据库.....	291	15.4	建立工具栏.....	348
13.3.1	认识数据库.....	292	15.4.1	建立 ToolBar 控件.....	348
13.3.2	建立数据库.....	292	15.4.2	工具按钮与程序代码结合.....	349
13.3.3	连接数据库与数据库控件.....	296	15.5	问题与解答.....	350
13.3.4	通过 ODBC 连接数据库.....	301	第 16 章	Windows API 与外部程序.....	353
13.3.5	使用 TDBEdit 控件 设计编辑窗口.....	306	16.1	Windows API 函数使用技巧.....	353
13.4	文件管理系统.....	308	16.1.1	认识 Windows API.....	353
13.4.1	读取文件与存储文件对话框.....	308	16.1.2	使用 Windows API 的函数.....	355
13.4.2	设置颜色对话框.....	310	16.2	API 函数应用举例.....	359
13.4.3	文件浏览程序.....	312	16.2.1	读取与设置计算机名称.....	359
13.5	问题与解答.....	315	16.2.2	用 API 函数读取.....	
第 14 章	数据打印技术.....	317			

鼠标坐标位置	360	18.2 猜猜看	409
16.2.3 使用 API 函数播放 多媒体文件	361	18.2.1 画面控件说明	409
16.3 API 绘图函数应用	364	18.2.2 程序代码应用技巧分析	410
16.3.1 用 API 函数绘制基本图形	364	18.2.3 互动设计与程序代码内容	411
16.3.2 用 API 绘图函数 设置图形的属性	365	18.3 配对游戏	415
16.4 调用外部程序	366	18.3.1 操作方法	416
16.4.1 用 WinExec 函数 调用外部程序	366	18.3.2 画面控件说明	416
16.4.2 调用控制台程序	368	18.3.3 程序代码应用技巧分析	417
16.5 问题与解答	368	18.3.4 互动设计与程序代码内容	417
第 17 章 网络程序设计与应用	370	18.4 配对记忆游戏	423
17.1 简易网络程序	370	18.4.1 操作方法	424
17.1.1 显示计算机的网络状态	370	18.4.2 画面控件说明	424
17.1.2 查询联机主机 IP 地址	372	18.4.3 互动设计与程序代码内容	424
17.2 简易在线通信程序	374	18.5 搬搬看	429
17.2.1 Winsock 使用的通讯协议	374	18.5.1 操作方法	430
17.2.2 使用 UDP 协议建立 联机程序	374	18.5.2 画面控件说明	430
17.2.3 使用 TCP 协议建立 联机程序	378	18.5.3 互动设计与程序代码内容	431
17.3 自制网络联机程序	382	18.6 角子老虎	438
17.3.1 自制浏览器	382	18.6.1 操作方法	439
17.3.2 查看网页的 HTML 原代码	384	18.6.2 画面控件说明	440
17.4 CGI 程序设计	386	18.6.3 互动设计与程序代码内容	440
17.4.1 建立 CGI 程序	386	18.7 智能盘	445
17.4.2 使用 CGI 程序显示数据表	389	18.7.1 操作方法	445
17.4.3 使用 CGI 程序查询数据表	395	18.7.2 画面控件说明	446
17.5 问题与解答	398	18.7.3 程序代码应用技巧分析	446
		18.7.4 互动设计与程序代码内容	447
		18.8 英文单词练习	451
		18.8.1 画面控件说明	451
		18.8.2 互动设计与程序代码内容	451
		18.9 踩地雷游戏	458
		18.9.1 操作方法	458
		18.9.2 画面控件说明	458
		18.9.3 程序代码应用技巧分析	459
		18.9.4 互动设计与程序代码内容	459
		18.10 问题与解答	467
		第 19 章 高互动型游戏专题	469
		19.1 神枪手	469
		19.1.1 操作方法	469
		19.1.2 画面控件说明	470

第IV部分 专题设计篇

第 18 章 智力游戏专题

18.1 拼图游戏	399
18.1.1 操作方法	400
18.1.2 画面控件说明	400
18.1.3 程序代码应用技巧分析	401
18.1.4 互动设计与程序代码内容	403

第 19 章 高互动型游戏专题

19.1 神枪手	469
19.1.1 操作方法	469
19.1.2 画面控件说明	470

19.1.3 互动设计与程序代码内容.....	470	19.4.2 画面控件说明.....	491
19.2 猴仔兵团	478	19.4.3 互动设计与程序代码内容	492
19.2.1 操作方法	478	19.5 青蛙过街.....	499
19.2.2 画面控件说明	479	19.5.1 操作方法.....	500
19.2.3 互动设计与程序代码内容.....	479	19.5.2 画面控件说明.....	500
19.3 钓鱼高手	484	19.5.3 互动设计与程序代码内容	501
19.3.1 操作方法	484	19.6 键盘操作练习.....	508
19.3.2 画面控件说明	485	19.6.1 操作方法.....	508
19.3.3 互动设计与程序代码内容.....	485	19.6.2 画面控件说明.....	509
19.4 打砖块	490	19.6.3 互动设计与程序代码内容	509
19.4.1 操作方法	491	19.7 问题与解答	517

第 I 部分 基本操作篇

第 1 章 C++ Builder 基本操作

Borland C++ Builder 是最知名的可视化 C++ 应用程序开发工具之一，它易学易用的操作界面和功能强大的应用组件，可以让程序员快速建立各种应用程序。所谓万丈高楼平地起，就让我们一起开始来认识 C++ Builder。

1.1 认识 C++ Builder 6

C++ Builder 是美国 Borland 公司最重要的程序语言产品之一，是功能强大而操作简单的 C++ 窗口程序开发工具。

1.1.1 认识 C++ Builder

1972 年 C 语言问世，C 语言编译器所产生出来的机器代码相当短，执行速度奇快，它能做到大部分汇编语言能做的事，是效率最高的高级语言，也有人称之为中级语言。在 UNIX 系统上，不论是何种品牌的机器，利用 C 与语言编写的程序，经过编译就能直接运行无误，即使是不同的操作系统，利用 C 语言编写的链接库也一样很容易跨平台转换，这使 C 语言成为有史以来最受欢迎的语言，也成为信息工业的标准语言，目前，几乎所有计算机专业的程序设计课程，都会将 C 语言列为必学的语言。

C++ 是将 C 语言加入面向对象概念的重要演进。C++ 语言成功地将 C 语言简洁、快速的优点与面向对象的软件工程设计方法相结合，成为主流的应用程序开发工具。

主要的 C++ 语言开发工具包括微软的 Visual C++、Borland 的 C++ Builder，以及 UNIX 系统中广泛使用的 GNU C++。

C++ Builder 是 Borland 公司于 1987 年推出的 C++ 程序开发工具，最新版本为 2002 年推出的 Borland C++ Builder 6(许多程序员将它简称为 BCB 6)。

C++ Builder 提供了一个集成开发环境 IDE，使程序员可以最少的程序代码来建立高效率的 Windows 应用程序。

C++ Builder 6 的版本分为 Enterprise(企业版)、Professional(专业版)和 Personal(个人版)3 个版本(本书以介绍专业版为主)，3 个版本的操作界面相同，主要的差异说明如下：

- Enterprise(企业版)：适用于企业开发人员、增值软件厂商、系统集成公司等需要建构现代 e-Business 应用系统的技术人员。
- Professional(专业版)：适用于需开发商业数据库应用程序、或想要进入新兴 LinuxR 市场的专业开发人员，可将 Windows 平台上的开发投资轻易地移植到 Linux 平台。
- Personal(个人版)：提供初学者以可视化开发工具，以鼠标拖放的方式，轻松学

习窗口程序设计，快速进入窗口程序设计领域，此版本仅授权供学习用途。

只要将 C++ Builder 6 的安装盘放入光驱中，安装程序就会自动引导用户将 C++ Builder 6 安装到硬盘中，安装完成专业版的开始菜单如图 1.1 所示。



图 1.1 安装 C++ Builder 6 后的【开始】菜单

另外，C++ Builder 提供了丰富完整的帮助文件，安装完成 C++ Builder 后，开始菜单中就会显示相关的帮助文件选项，如图 1.2 所示。

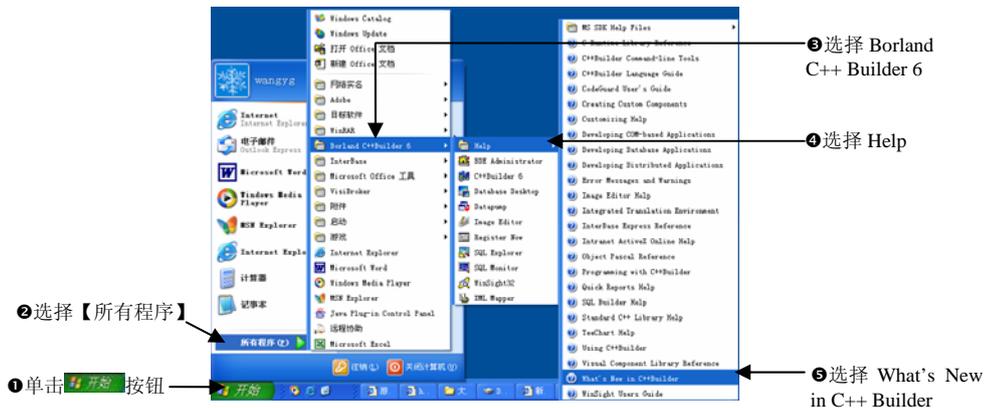


图 1.2 C++ Builder 的帮助文件选项

弹出【帮助主题】对话框，可以在此对话框中查阅说明文件的内容，如图 1.3 所示。



图 1.3 【帮助主题】对话框

1.1.2 认识 C++ Builder 的窗口

激活 C++ Builder 后, 将发现窗口中有许多子窗口, 每一个子窗口分别负责不同的工作, 而为了不同的工作需求, 可以随时打开或关闭子窗口。以下就是打开子窗口的操作示范:



示范

在菜单栏选择 View。

从弹出的菜单中选择想要打开的子窗口名称, 如图 1.4 所示。

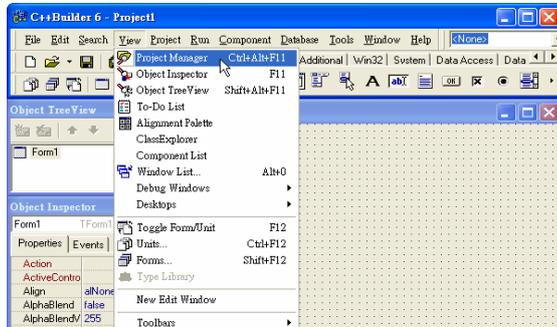


图 1.4 选择子窗口名称

例如, 选择 Project Manager 命令, 就会激活 Project Manager 窗口, 如图 1.5 所示。

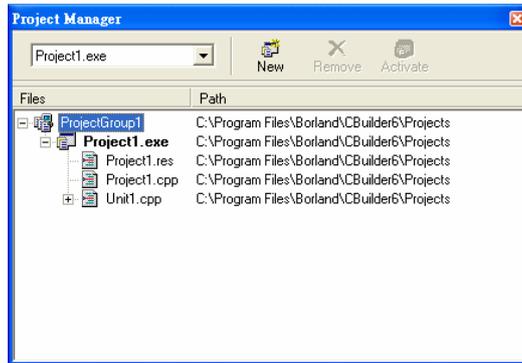


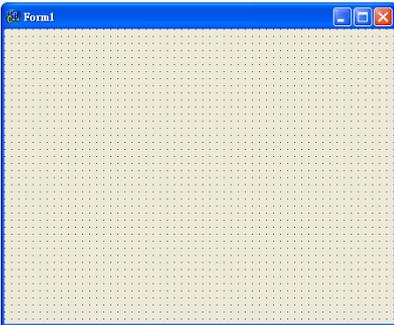
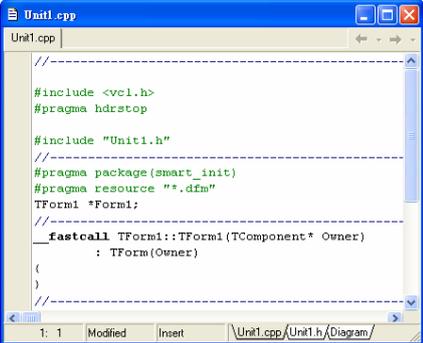
图 1.5 Project Manager 窗口

各子窗口的界面与功能如表 1.1 所示。

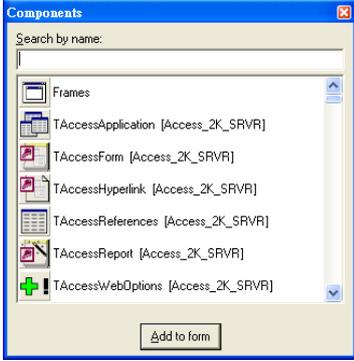
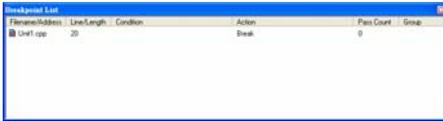
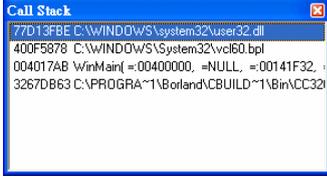
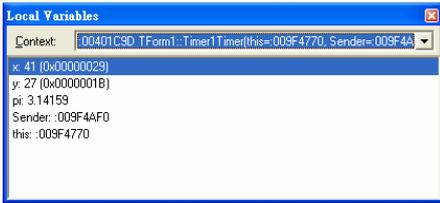
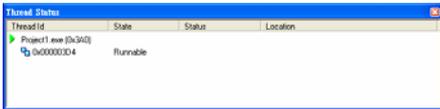
表 1.1 C++ Builder 的子窗口

窗口名称	窗口界面	主要功能
Project Manager (项目管理器)		用来管理窗体、模块等, 包括新建、打开、保存、删除等功能

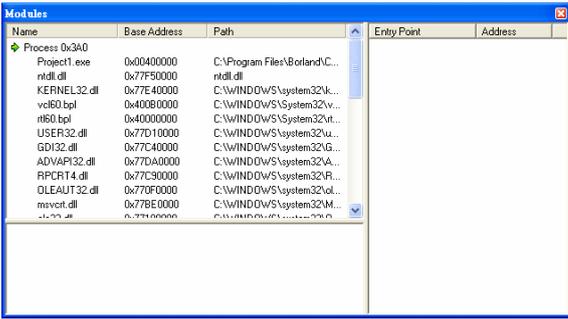
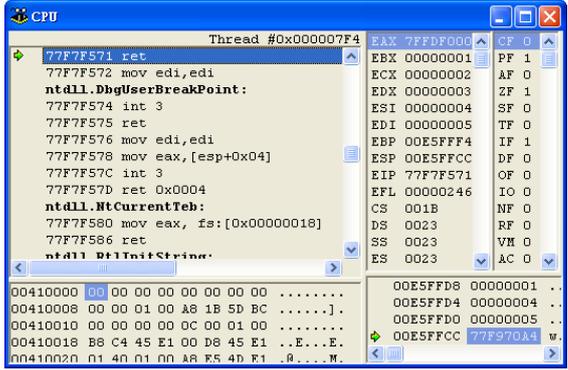
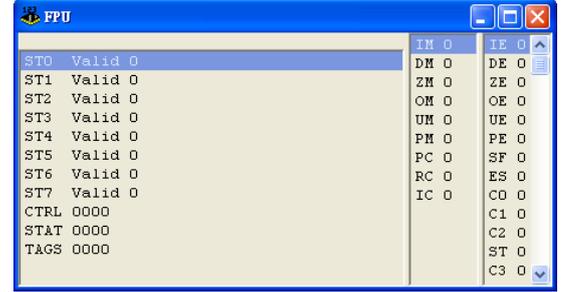
续表

窗口名称	窗口界面	主要功能
Form (窗体)		用来安排程序操作界面的窗口,在界面上要显示的组件都要放到窗体中
Object Inspector (属性窗口)		用来设置各种组件的属性,例如:属性名称、组件位置、大小等
Unit (程序代码窗口)		用来编写程序语言的窗口
ClassExplorer (类浏览器)		通常与 Unit 集成在一起,可以迅速浏览程序代码所在的位置

续表

窗口名称	窗口界面	主要功能
Align (对齐设置面板)		用来设置组件间的对齐方式
Components (组件)		查看目前可以使用的组件项目，可以直接加入窗体中
Breakpoint List (断点)		显示所有断点状态的窗口
Call Stack (调用堆栈)		用来查看目前的程序堆栈情况
Watch List (监看窗口)		可以用来监看或追踪变量的变化情况
Local Variables (区域变量窗口)		可以用来查看变量的变化情况
Thread Status (执行状态窗口)		可以用来监看程序的执行状态

续表

窗口名称	窗口界面	主要功能
Modules (模块窗口)		显示相关模块内容
CPU (CPU 指令窗口)		显示 CPU 中的执行程序代码
FPU (缓存器状态窗口)		显示缓存器的值与状态

1.1.3 认识工具按钮

除了子窗口以外，C++ Builder 主窗口上方有 3 个工具栏——Standard、View 和 Debug，其中放置了一些经常使用的工具按钮，如图 1.6 所示。



图 1.6 Standard、View 和 Debug 工具栏

工具栏上主要工具按钮的功能如表 1.2 所示。

表 1.2 主要工具按钮的功能

按钮	名称	功能说明
	New	用来新建各种项目(item), 包括项目、窗体、ActiveX、对话框等
	Open	打开在硬盘中已存在的文件
	Save	将正在工作的项目保存起来, 包括相关的窗体与模块
	Save All	将所有项目保存起来, 包括相关的窗体与模块
	Open Project	打开在硬盘中已存在的项目
	Add file to project	将文件加入工作项目中
	Remove file from project	删除工作项目中的文件
	View Unit	选择想要查看的程序代码窗口
	View Form	选择想要编辑的窗体
	Toggle Form/Unit	切换窗体与对应的程序代码窗口
	New Form	打开一个新的窗体
	Run	编译程序并开始执行
	Pause	暂停程序执行
	Trace into	每次执行一行
	Step over	每次执行一行(function call 除外)

1.2 建立一个项目

对生手来说, 如何开始建立一个新的项目是非常重要的, 本节将逐步引导用户学习建立项目的相关技巧与观念。

1.2.1 建立新项目

项目(Project)是用来管理所有相关窗体、模块的文件, 在开始着手开发新的应用程序时, 必须先建立一个项目文件(扩展名为.bpr), 然后逐步在项目上增加各种窗体和程序模块。项目文件的图标如图 1.7 所示。

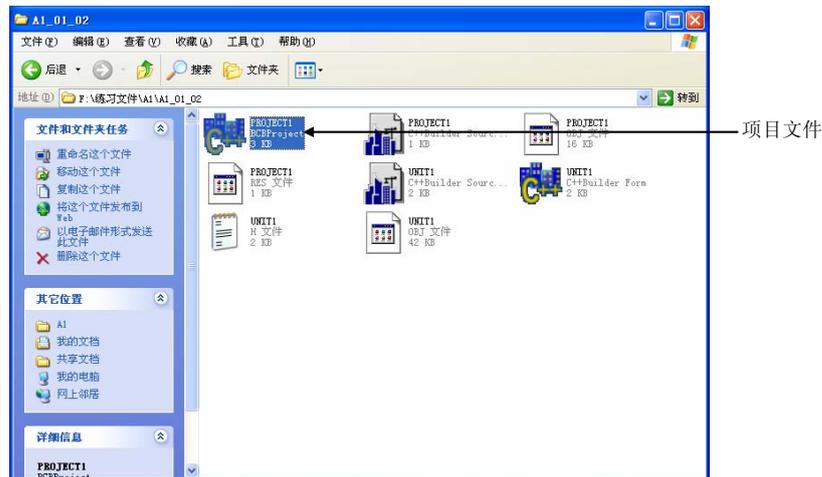


图 1.7 项目文件的图标

建立新的项目时，最好先建立一个这个项目专用的文件夹，以免不同项目使用了相同的文件名而造成混淆。

激活 C++ Builder 时，会自动建立新项目，用户可以直接使用这个新项目来编辑程序，也可以自行建立新的项目。以下就是建立新项目的操作示范：



在菜单栏选择 File。

从弹出的菜单中选择 New。

从子菜单中选择 Other，如图 1.8 所示。

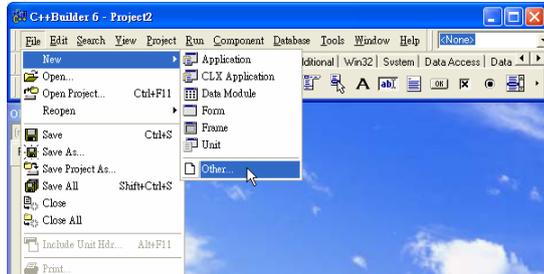


图 1.8 选择新建项目的命令



弹出 New Items 窗口后，选择 Application。

单击 按钮，如图 1.9 所示。

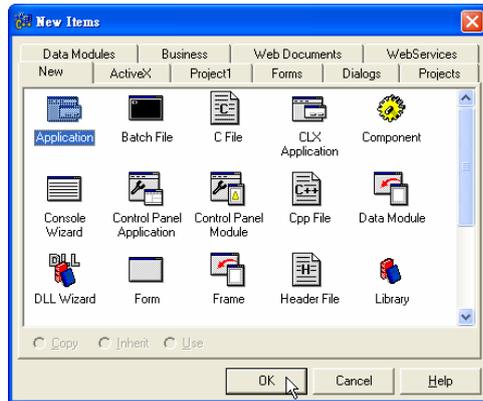
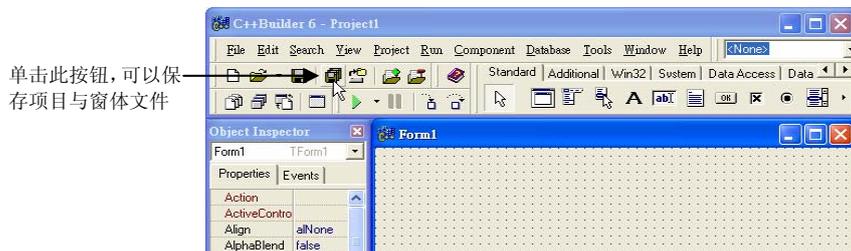


图 1.9 选择项目类型

建立新项目后，会自动产生一个窗体(Form)，单击 Save All 按钮，就可以将项目文件和窗体文件保存起来，如图 1.10 所示。



单击此按钮，可以保存项目与窗体文件

图 1.10 保存项目文件和窗体文件

1.2.2 打开项目文件

使用打开项目功能,可以将保存在硬盘中的项目文件以及相关窗体和模块文件加载到 C++ Builder 窗口中。以下就是打开项目的操作示范:



在菜单栏选择 File。

从弹出的菜单中选择 Open Project 命令,如图 1.11 所示。

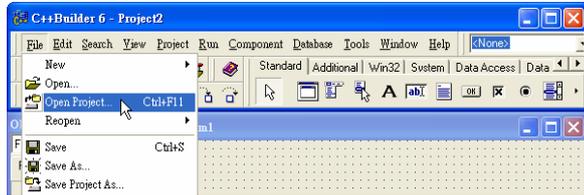


图 1.11 选择打开项目的命令



弹出 Open Project 对话框后,选择文件保存的文件夹。

选择项目文件。

单击 按钮,如图 1.12 所示。

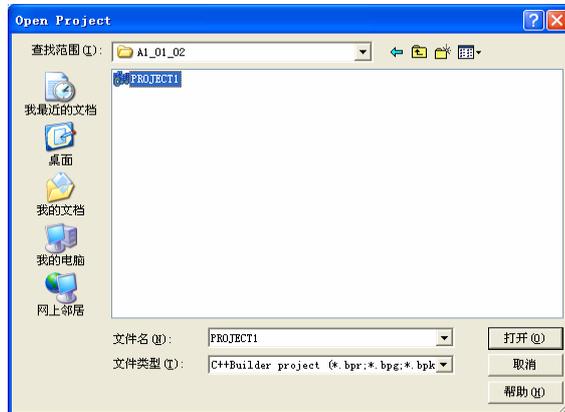


图 1.12 选择要打开的项目文件

打开项目后,如图 1.13 所示。

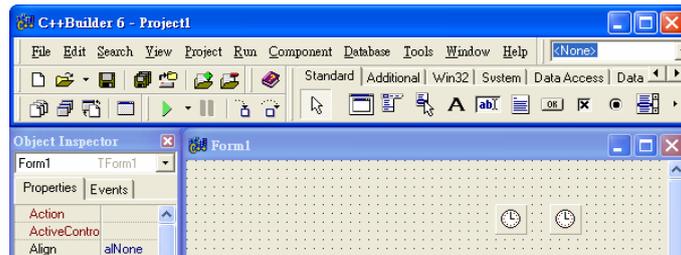


图 1.13 打开的项目文件



重要

本书附书光盘中的练习文件和验证文件都是只读文件，为了方便练习，最好先将文件复制到硬盘中，并取消只读属性。

把文件复制到硬盘后，取消只读属性的方法是：在该文件上右击，然后从弹出的快捷菜单中选择【属性】命令，弹出【属性】对话框，取消选中【只读】复选框，如图 1.14 所示。



图 1.14 【Project1 属性】对话框

为了增进学习的成效，练习每一个章节时，建议用户先打开该章节的练习文件，然后再开始操作。

1.2.3 新建与删除窗体

窗体(Form)是专门用来设计操作界面的窗口，所有要呈现在界面上的数据都要安排在窗体上。在一个项目上可以包含多个窗体，利用程序来切换想要显示的窗体内容。以下就是新建窗体的操作示范：

请先打开练习文件 A1_02_03 的 Project1.bpr

示范

在菜单栏选择 File。

从弹出的菜单中选择 New。

从子菜单中选择 Form，如图 1.15 所示。

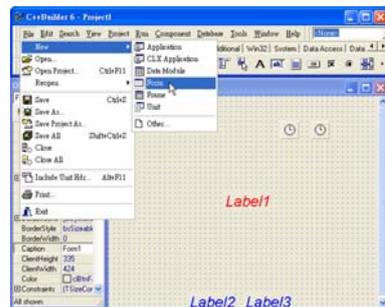


图 1.15 选择新建窗体的命令

新建窗体后，如图 1.16 所示。

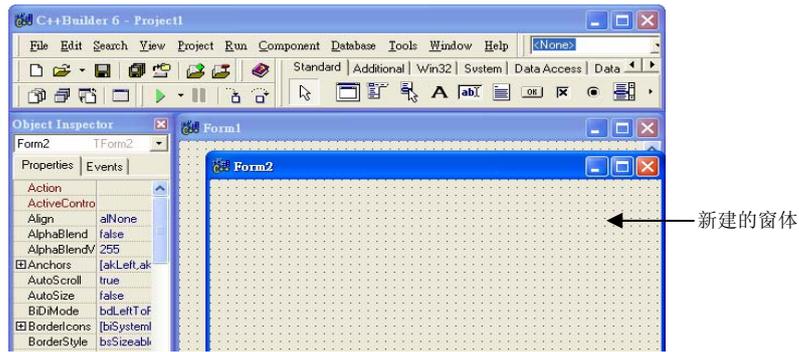


图 1.16 新建的窗体

新建窗体后，选择 Save All 按钮，就会建立一个新的窗体文件，而窗体相关的扩展名为 .cpp、.dfm、.h，如图 1.17 所示。

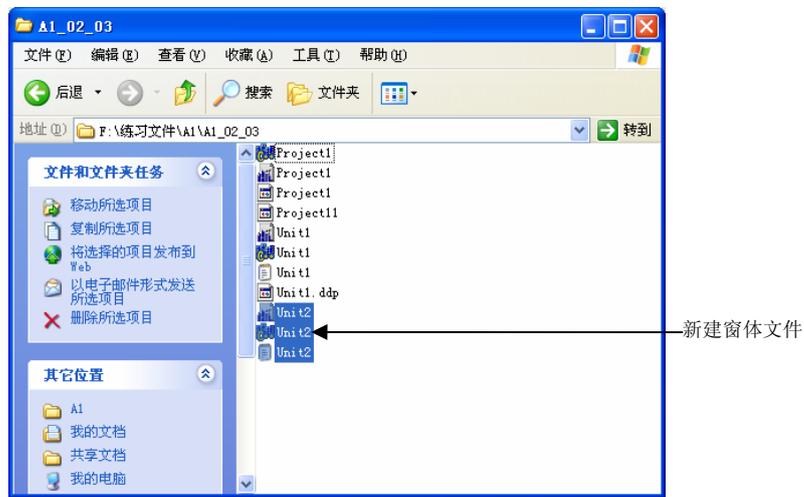


图 1.17 新建的窗体文件

新建的窗体名称会显示在 Project Manager 窗口中，可以在 Project Manager 窗口中直接将窗体删除。以下就是删除窗体的操作示范：



单击工具栏的 Remove file from project 按钮，如图 1.18 所示。

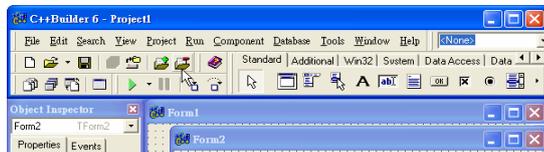


图 1.18 单击 Remove file from project 按钮



弹出 Remove From Project 窗口后，选择想要删除的窗体名称。单击 **OK** 按钮，如图 1.19 所示。

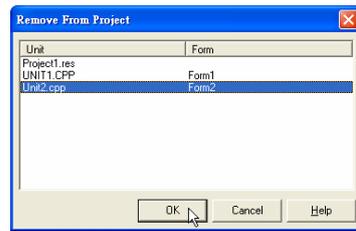


图 1.19 选择要删除的窗体

将 Project Manager 窗口中的窗体删除后，硬盘中的窗体文件并不会真正地自动删除，如果再使用新建窗体的打开功能，又可以将窗体加入到这个项目中。

1.2.4 在窗体上新建组件

C++ Builder 主窗口上方的 Component Palette 工具按钮，就是用来建立各种组件(控件)的工具，下面就来实际体验一下，如何在窗体上建立组件。以下就是在窗体上新建组件的操作示范：

请先打开练习文件 A1_02_04 的 Project1.bpr



选择 Component Palette 的 Standard 标签。

单击 **A** Label 按钮。

在窗体上按住鼠标左键拖动，如图 1.20 所示。

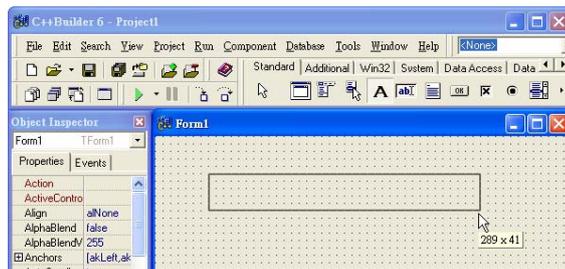


图 1.20 新建组件



在 Object Inspector 窗口的 Caption 属性值栏输入想要显示的文字，如图 1.21 所示。

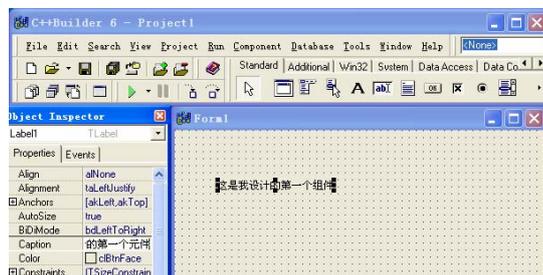


图 1.21 设置组件的属性

1.3 编写与测试程序

前面介绍了许多基本操作技巧，接下来介绍编写程序和执行程序的过程和方法。

1.3.1 在程序代码窗口输入程序代码

在每一个项目中可以建立多个窗体(Form)，且窗体中可以分别置入各种组件。每一个窗体会分别对应一个程序代码窗口，在程序代码窗口中可以输入与窗体或组件有关的程序代码。以下就是打开程序代码窗口的操作示范：

示范 1

单击工具栏的  Toggle Form/Unit 按钮，切换窗体与程序代码窗口，如图 1.22 所示。

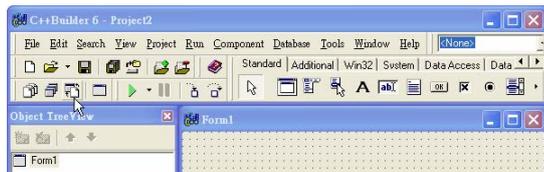


图 1.22 单击 Toggle Form/Unit 按钮

示范 2

在 Object Inspector 窗口中选择 Events 标签。

在 OnActivate 属性值栏中双击，如图 1.23 所示。

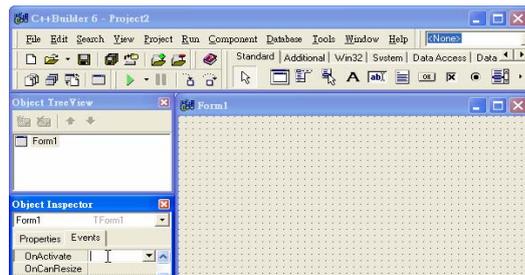


图 1.23 设置 OnActivate 属性栏

示范 3



菜单栏选择 View。

从弹出的菜单中选择 Units 命令，如图 1.24 所示。

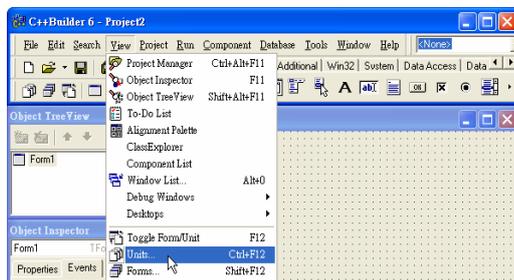


图 1.24 选择 Units 命令



弹出 View Unit 窗口后，选择 Unit1。
单击 按钮，如图 1.25 所示。

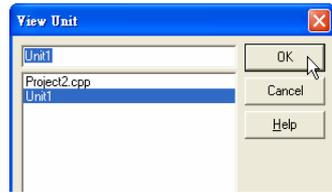


图 1.25 选择要打开的窗体

打开程序代码窗口，如图 1.26 所示。

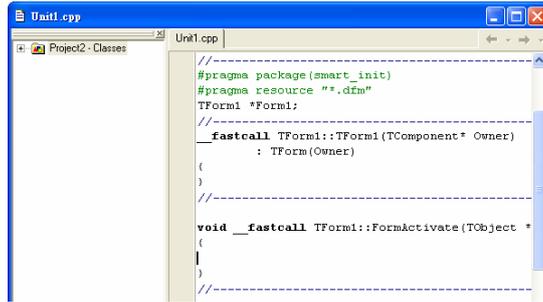


图 1.26 程序代码窗口

采用示范 2 的方法，除了打开程序代码窗口外，还会出现如下的程序代码：

```
void __fastcall TForm1::FormActivate(TObject *Sender)
{
}

```

Activate 是事件程序的一种，在窗体发生作用时，会发生的事件，进而执行 FormActivate 中的程序代码。（关于事件程序的详细说明请参考第 11 章的内容。）

在学习初期，可以使用 Activate 事件来练习简单的程序代码。以下就是使用 Activate 事件来执行程序代码的操作示范：



请先打开练习文件 A1_03_01 的 Project1.bpr



在 Object Inspector 窗口中选择 Events 标签。

在 OnActivate 属性值栏中双击，如图 1.27 所示。

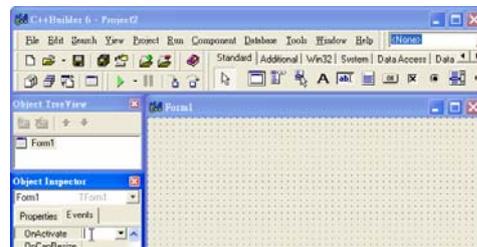


图 1.27 设置 OnActivate 属性栏



弹出 Unit1.cpp 窗口后，在窗口中输入程序代码，如图 1.28 所示。

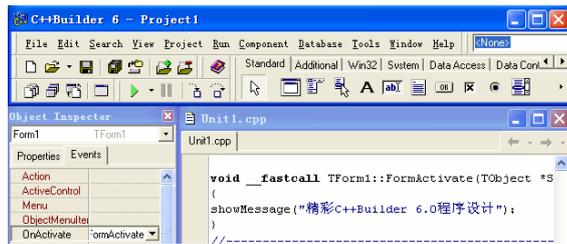


图 1.28 Unit1.cpp 窗口中输入程序代码

完成后，单击工具栏的 Run 按钮，执行结果如图 1.29 所示。



图 1.29 程序执行结果

虽然单击窗体右上角的【关闭窗口】按钮，可以结束程序的执行，不过经常无法正常回到 C++ Builder 窗口，最佳结束程序执行的方法，是从菜单栏的 Run 菜单中选择 Program Reset 命令来结束程序的执行(或按快捷键 Ctrl+F2)，如图 1.30 所示。

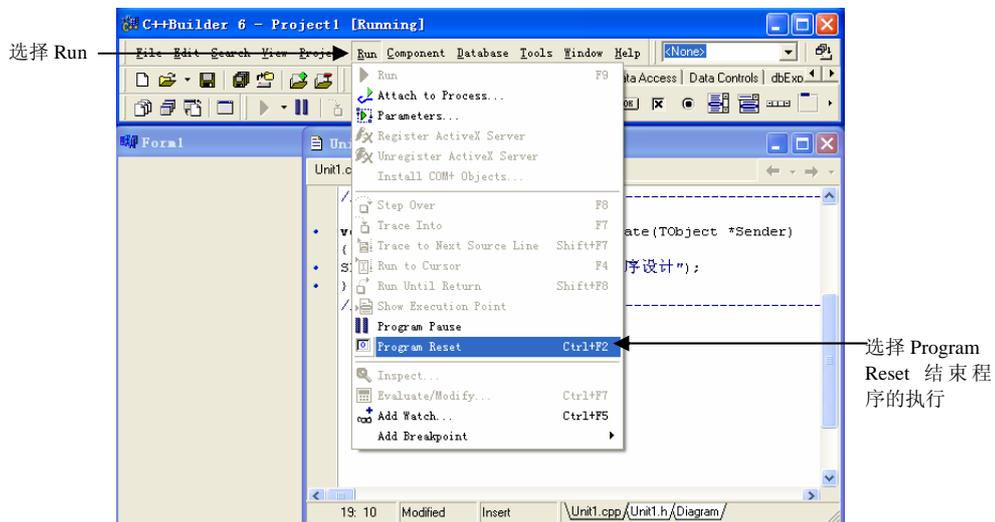


图 1.30 选择 Program Reset 命令来结束程序的执行

1.3.2 使用按钮来控制程序的执行

下面这个例子，要说明如何在窗体上设计一个按钮，当单击按钮时，才开始执行程序代码。以下就是使用按钮控制程序执行的操作示范：

 请先打开练习文件 A1_03_02 的 Project1.bpr



选择 Component Palette 的 Standard 标签。

选择  Button 按钮。

在窗体上按住鼠标左键拖动，如图 1.31 所示。

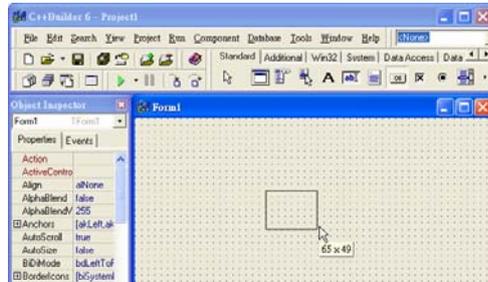


图 1.31 新建按钮组件



在 Button1 组件上双击，如图 1.32 所示。

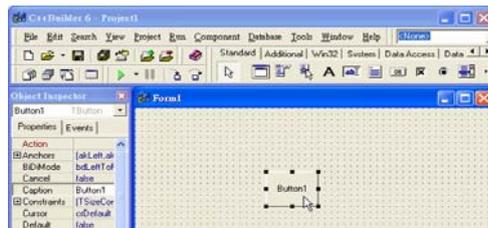


图 1.32 设置组件属性



弹出 Unit1.cpp 窗口后，输入程序代码，如图 1.33 所示。

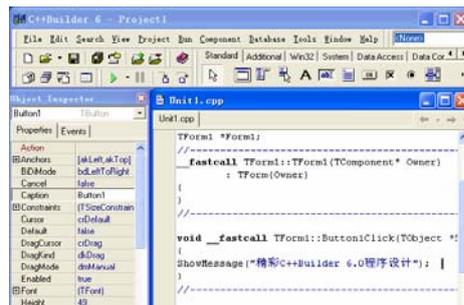


图 1.33 输入程序代码

完成后，单击工具栏的  Run 按钮，然后在设计的按钮上单击，就会执行程序代码的内容，如图 1.34 所示。

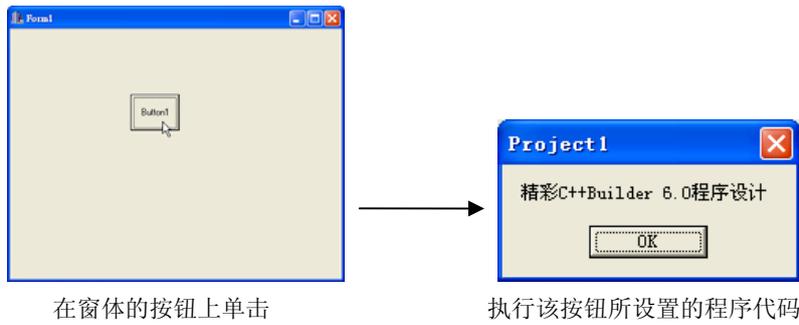


图 1.34 执行程序结果

1.3.3 输入组件与计算结果

只要在窗体上安排输入组件和执行按钮，然后输入简单的程序代码，就可以建立一个简单的数字系统转换程序。以下就是制作“输入组件与计算结果”的操作示范：

请先打开练习文件 A1_03_03 的 Project1.bpr



选择 Component Palette 的 Standard 标签。

单击 Edit 按钮。

在窗体上按住鼠标左键拖动，如图 1.35 所示。

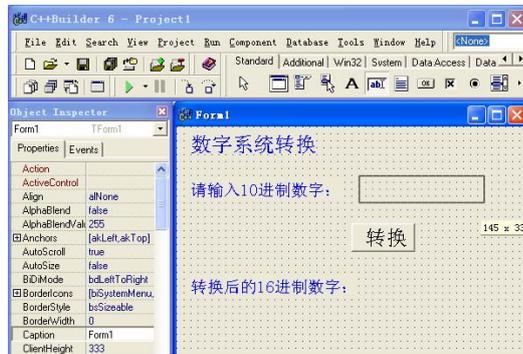


图 1.35 建立一个 Edit 组件



重复步骤 ，再建立一个 Edit 组件，如图 1.36 所示。

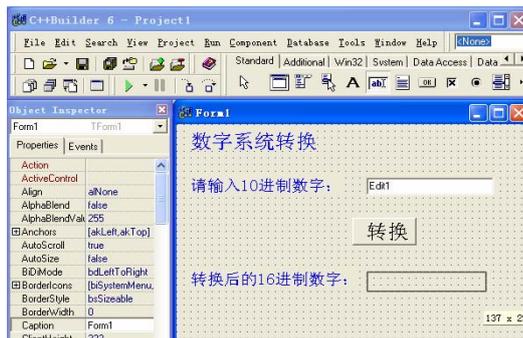


图 1.36 建立一个 Edit 组件



在 Button1 组件上双击, 如图 1.37 所示。

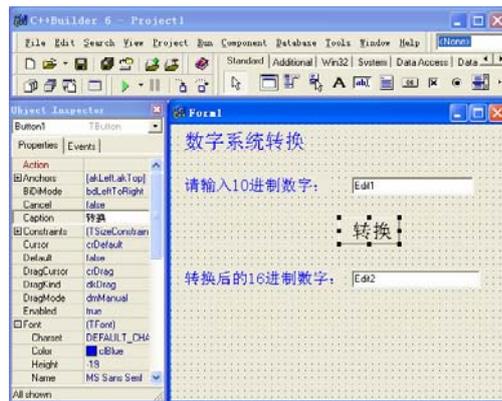


图 1.37 双击 Button1 组件



弹出 Unit1.cpp 窗口后, 输入数字系统转换的程序代码, 如图 1.38 所示。

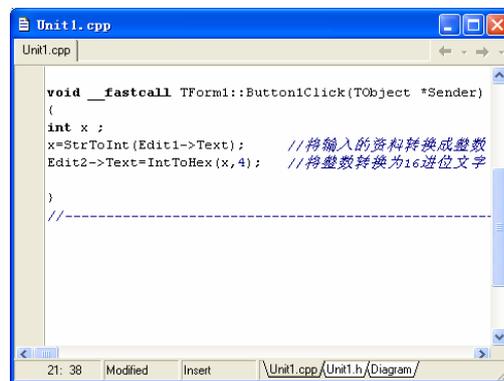


图 1.38 输入数字系统转换程序代码

完成后, 单击工具栏的 Run 按钮, 然后在 Edit1 组件中输入要转换为 16 进制的数字, 再单击 转换按钮, 就可以在 Edit2 组件中看到转换后的结果, 如图 1.39 所示。

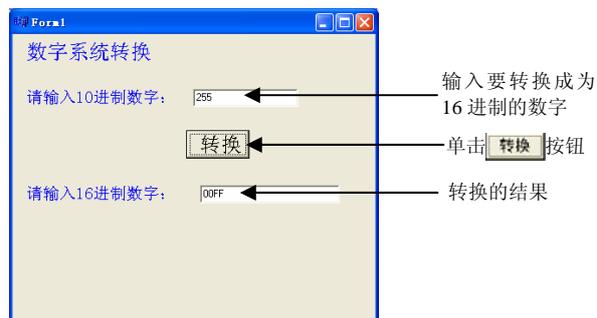


图 1.39 转换的结果

在  中输入的程序代码解析如下：

```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    int x ;
    x=StrToInt(Edit1->Text);    //将输入的数据转换成整数
    Edit2->Text=IntToHex(x,4);  //将整数转换为 16 进制文字
}
```

- Edit1、Edit2 分别为在窗体上新建的文字盒名称，而 Edit1->Text 和 Edit2->Text 分别代表这两个文字盒中存放的文字内容属性。
- `x = StrToInt(Edit1->Text);`
此段程序代码中，StrToInt 函数会将 Edit1->Text 的文字内容转换为数值，并指定给 x 变量。
- `Edit2->Text=IntToHex(x,4);`
此段程序代码中，IntToHex 函数会将十进制的 v1 数值转换为 16 进制，并指定给 Edit2 的 Text 属性。

1.3.4 程序代码窗口的辅助编辑功能

在程序代码窗口中设置组件的属性或使用组件的方法时，会自动弹出相关的属性和方法，以帮助用户使用正确的名称，如图 1.40 所示。



图 1.40 程序代码窗口属性和方法

在组件名称的后面加上“->”符号后，就会自动显示属性与方法菜单。以下就是应用辅助编辑菜单的操作示范：

 请先打开练习文件 A1_03_04 的 Project1.bpr



在 Object Inspector 窗口中选择 Events 标签。

在 OnActivate 属性值栏中双击，如图 1.41 所示。

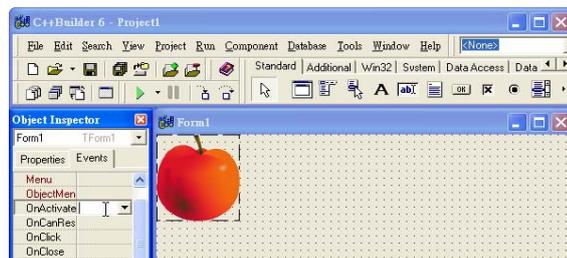


图 1.41 设置 OnActivate 属性



弹出 Unit1.cpp 窗口后，输入组件名称与“->”符号，如图 1.42 所示。

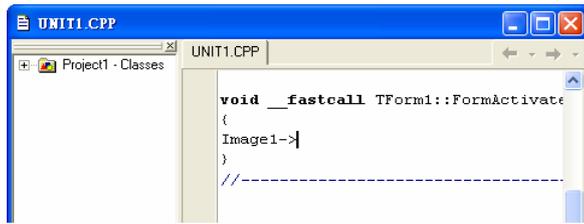


图 1.42 输入组件名称



弹出相关属性菜单后，可以从菜单中查询想要输入的属性与方法，依照菜单提供的数据，输入组件的属性与使用方法，如图 1.43 所示。

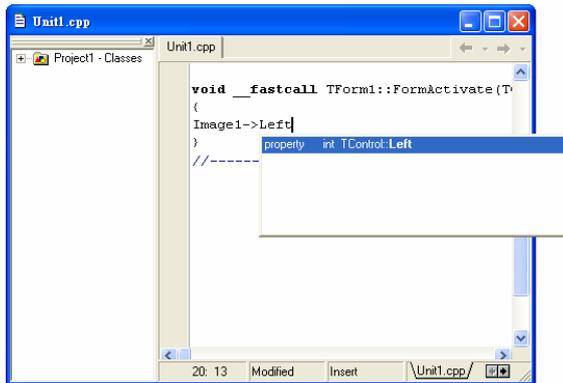


图 1.43 属性菜单



程序代码输入完毕后，输入“;”符号结尾，菜单就会自动消失，如图 1.44 所示。

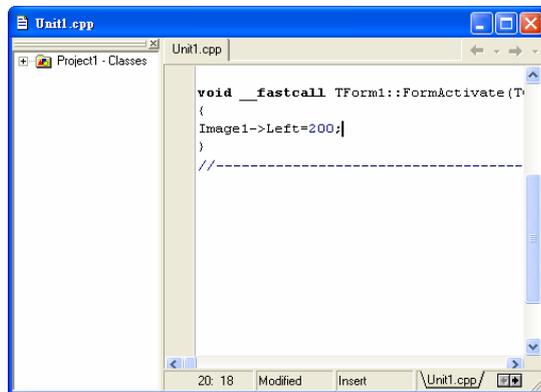


图 1.44 程序代码输入完毕

有了组件属性和使用方法的辅助菜单，在输入程序代码时就更方便了。

1.4 问题与解答

1. 回答下列问题。

- (1) C++ Builder 项目的扩展名为()。
- (2) C++ Builder 程序的扩展名为()。
- (3) C++ Builder 窗体的扩展名为()。
- (4) C++ Builder 定义文件的扩展名为()。
- (5) C++ Builder 最新版本为()。

 重点提示：请参考 1.2 节。

2. 尝试编写一段程序代码，以显示如图 1.45 所示的结果。



图 1.45 简易加法器执行界面

 重点提示：请参考 1.3 节。