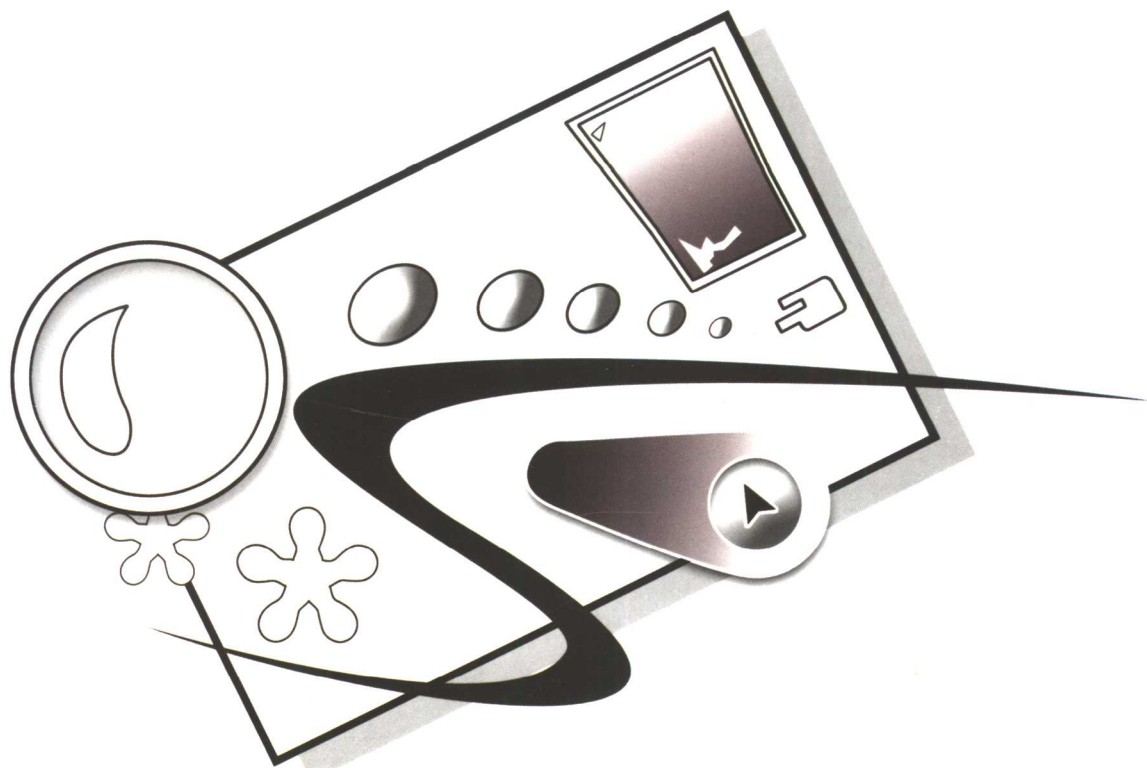


高等职业教育计算机专业推荐教材



Delphi

程序设计教程



网上免费提供
实例源代码



夏云龙 张桂芸 于晓华 等编著

高等职业教育计算机专业推荐教材

Delphi 程序设计教程

夏云龙 张桂芸 于晓华 等编著



机械工业出版社

本书共分 13 章,由浅入深地介绍了 Delphi 程序设计方法。首先,介绍了 Delphi 的开发环境,并通过一个简单的程序示例帮助读者了解开发 Delphi 程序的一般步骤;然后,系统地介绍了 Delphi 语言的基本知识、窗体的设计方法、组件的使用方法和代码书写标准等;最后,重点讲解了一些实用程序的开发方法,这些程序包括 MDI 应用程序、多媒体程序、文件处理程序、系统程序、数据库以及网络程序等。

本书内容全面、概念清晰、深入浅出、通俗易懂、实用性强。适合作为高职高专院校相关专业的 Delphi 语言教材,也可以作为 Delphi 开发人员的参考书。

图书在版编目(CIP)数据

Delphi 程序设计教程/夏云龙等编著. —北京:机械工业出版社,2006.6

(高等职业教育计算机专业推荐教材)

ISBN 7-111-18807-1

I. D... II. 夏... III. 软件工具—程序设计—高等学校:技术学校—教材 IV. TP311.56

中国版本图书馆 CIP 数据核字(2006)第 027597 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

策 划:胡毓坚

责任编辑:陈振虹

责任印制:李 妍

北京中兴印刷有限公司印刷

2006 年 5 月第 1 版·第 1 次印刷

787mm×1092mm 1/16·19.25 印张·477 千字

0001—5000 册

定价:27.00 元

凡购本图书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话:(010)68326294

编辑热线:(010)88379739

封面无防伪标均为盗版

高等职业教育计算机专业推荐教材

编委会成员名单

主 任 王元元

编 委 丁跃潮 黄陈蓉 黄国兴
李咏梅 逯燕玲 王爱梅
奚李峰 杨世平 张桂芸

编者的话

根据有关部门对我国信息产业发展的客观需求及劳动力市场现状的调查,在计算机应用和软件专业领域培养技能型紧缺人才,是当务之急。近年来,不仅高等职业技术学院,而且相当一部分本科工程技术院校(特别是相当数量高等学校的二级学院、民办院校),都把招收和培养计算机专业技能型紧缺人才列为教育改革的重要举措。为一些院校提供“适时、适度、优选、优质”的计算机专业的高等职业教育系列教材,正是我们组织编写这套“高等职业教育计算机专业推荐教材”(以下简称“推荐教材”)的目的。“推荐教材”由四个模块的30多本教材组成。这些模块是:基础知识模块、程序设计模块、实用技术模块、实践模块。

这套“推荐教材”是“适时”的,因为它努力适应我国信息产业发展和劳动力市场的客观需求,适应计算机行业技术的现状,强调教学内容的先进性和实用性。这套教材十分注意关注信息技术的最新发展,突出本专业领域的新知识、新技术、新流程和新方法。其中程序设计模块和实用技术模块充分体现了这一特色,所涉及的19本教材既有基础的平台、语言,如《Linux操作系统》、《C语言程序设计与实践》,也有最新的《Visual C#.NET 面向对象程序设计教程》、《XML 实用教程》、《JSP 应用教程》等工具,还有十分接近实际工作需要的《Oracle 数据库应用教程》、《计算机网络管理》、《电子商务概论》等实用教材。

这套“推荐教材”是“适度”的,因为它不是简单地摒弃基础理论,而是注意强调理论联系实际,使读者能从中学习到必要和相对系统的基础理论知识,把各种能力的培养和全面素质的提高放在首要的位置。“推荐教材”中基础知识模块的设置,充分体现了这一特色,它包括了数学、电子技术、计算机硬件、软件和应用技术、网络技术、信息安全等基础教材。

这套“推荐教材”是“优选”的,因为它充分考虑了现有高中毕业生的认知水平和已有知识,为学生提供适应劳动力市场需要和有职业发展前景的、模块化的教材体系。在学习内容、教学组织等方面留给教师和学生选择和创新的空間,便于教师组织和构建开放式的课程体系,适应学生个性化发展的需要,在灵活的模块化课程结构中自由发展。“推荐教材”的四个模块对重要内容都安排了看似重复的多种教材,供教师和学生去选择。例如,可以在《C语言程序设计与实践》、《Visual Basic.NET 可视化程序设计教程》、《Visual C#.NET 面向对象程序设计教程》、《Java 程序设计教程》中任意选择一到两门,也可以在《ASP 基础及应用教程》、《JSP 应用教程》中任选一门。

这套“推荐教材”是“优质”的,因为它们的作者多数是从事高等职业教育的计算机专业教师,具有长期的计算机实际工作和教育工作经验。这套教材的优质,还体现在它的改革和创新精神上。其中《计算机电路基础》对传统的模拟电路和数字电路课程教材作了重大的改变;《计算机组装与维修教程》则是一门纯实践的课程教材。我们欢迎使用这套教材的师生,指出教材中存在的问题并提出修改意见。

高等职业教育计算机专业推荐教材

编委会

前 言

Delphi 是一套相当强大的工具,同时它也是一套复杂的程序设计环境。该环境中包含着大量元素,例如,它提供了数据库功能、客户/服务器功能、内联网功能和互联网解决方案,它还提供了强大的控制和管理功能,大大提高了软件的生产效率。

本书从 Delphi 的一个小程序例子开始全面系统地讲述了 Delphi 语言的基本知识和多种不同类型应用程序的开发方法。首先,介绍了 Delphi 的开发环境,并通过一个简单的程序示例帮助读者了解开发 Delphi 程序的一般步骤;然后,系统地介绍了 Delphi 语言的基本知识、窗体的设计方法、组件的使用方法和代码书写标准等;最后,重点讲解了一些实用程序的开发方法,这些程序包括 MDI 应用程序、多媒体程序、文件处理程序、系统程序、数据库以及网络程序等。最后,在附录中详细列出了在开发 Delphi 程序时大部分常见错误的中文注释。在叙述上,本书力求做到概念清晰、通俗易懂。

为了更好地使读者理解并掌握相关程序的开发方法和技巧,本书使用了大量的示例代码。在对 Delphi 相关理论进行介绍之后,本书都将使用一个完整的具体实例来演示这些功能是如何被应用的。

就 Delphi 程序员来说,本书是一本很好的辅助工具书。正在学习或使用 Delphi 开发应用程序的读者,也可以从本书中得到重要的帮助。

本书第 1 章由周曦国编写,第 2~4 章由张桂芸编写,第 5~8 章由于晓华编写,第 9~13 章由夏云龙编写。

本书实例源代码可在机械工业出版社网站(<http://www.cmpbook.com>)上下载。

由于编者水平有限,错误和不妥之处,请读者批评指正。

编者

目 录

编者的话

前言

第 1 章 初识 Delphi	1
1.1 Delphi 简介	1
1.2 集成开发环境(IDE)	1
1.2.1 IDE 概述	2
1.2.2 菜单栏	2
1.2.3 工具栏	3
1.2.4 组件面板	3
1.2.5 Form Designer	3
1.2.6 Object Inspector	4
1.2.7 Object TreeView	6
1.2.8 Code Explorer	6
1.3 第一个 Delphi 程序	7
1.3.1 设计程序界面	7
1.3.2 编写程序代码	10
1.3.3 进一步的讨论	14
1.4 小结	15
1.5 习题	15
第 2 章 数据类型与运算符	17
2.1 程序结构	17
2.1.1 项目文件结构	17
2.1.2 单元文件结构	18
2.1.3 单元引用和 uses 子句	19
2.1.4 包文件结构	22
2.2 基本语法元素	22
2.2.1 特殊符号	23
2.2.2 标识符	23
2.2.3 限制标识符	24
2.2.4 保留字	24
2.2.5 指示字	25
2.2.6 数字	25
2.2.7 标号	25
2.2.8 字符串	25
2.2.9 注释和编译指示	26
2.3 变量与常量	27
2.3.1 常量	27

2.3.2 变量	29
2.4 数据类型	31
2.4.1 简单数据类型	31
2.4.2 串类型	38
2.4.3 结构类型	40
2.4.4 指针和指针类型	50
2.4.5 变体类型	52
2.4.6 类型兼容和等同	55
2.4.7 强制类型转换	57
2.5 运算符与表达式	58
2.5.1 算术运算符	59
2.5.2 关系运算符	60
2.5.3 布尔运算符	61
2.5.4 位运算符	62
2.5.5 串运算符	62
2.5.6 集合运算符	63
2.5.7 指针运算符	63
2.5.8 地址运算符	64
2.5.9 类运算符	64
2.5.10 运算符优先规则	65
2.6 小结	65
2.7 习题	66
第3章 语句与例程	67
3.1 语句	67
3.1.1 简单语句	67
3.1.2 结构语句	68
3.1.3 块和作用域	77
3.2 例程	79
3.3 参数	87
3.3.1 参数的定义	88
3.3.2 串参数	91
3.3.3 数组参数	91
3.3.4 默认参数	94
3.4 调用例程	95
3.5 常用例程	96
3.5.1 处理空结束串例程	96
3.5.2 其他常用例程	97
3.6 小结	100
3.7 习题	100
第4章 面向对象编程	102
4.1 类和对象	102

4.2 类类型	103
4.2.1 TObject	103
4.2.2 类类型的兼容性	104
4.2.3 类成员的可见度	104
4.2.4 前置声明和相互依赖的类	106
4.3 域	106
4.4 方法	107
4.4.1 方法的声明和实现	107
4.4.2 binding	109
4.4.3 reintroduce	111
4.4.4 overload	112
4.4.5 构造器和析构器	112
4.4.6 类方法	115
4.5 属性	115
4.5.1 属性访问	116
4.5.2 数组属性	117
4.5.3 索引说明符	119
4.5.4 存储说明符	119
4.5.5 属性覆盖和再声明	120
4.6 类引用	121
4.6.1 类引用的类型	121
4.6.2 构造器和类引用	122
4.6.3 类运算符	122
4.7 小结	123
4.8 习题	123
第5章 异常	125
5.1 使用异常	125
5.1.1 声明异常类型	126
5.1.2 引发异常	126
5.2 处理异常	127
5.2.1 try...except 语句	127
5.2.2 重触发异常	129
5.2.3 嵌套异常	130
5.2.4 try...finally 语句	131
5.3 执行流程	132
5.4 小结	134
5.5 习题	134
第6章 窗体与组件	135
6.1 窗体	135
6.1.1 优秀的窗体设计	135
6.1.2 创建可重用的窗体	136

6.1.3 窗体间的相互引用	139
6.1.4 窗体的内部结构	143
6.1.5 窗体的属性、方法和事件	145
6.2 组件	149
6.2.1 组件的概念	149
6.2.2 组件的分类	150
6.3 组件基本操作	150
6.3.1 向窗体中添加组件	150
6.3.2 删除窗体中的组件	151
6.3.3 调整组件的大小和位置	151
6.3.4 设置可视组件的 Tab Order	152
6.3.5 组件的锁定与解锁	153
6.3.6 设置窗体中组件的属性	153
6.3.7 创建组件的事件处理过程	153
6.3.8 其他操作	153
6.4 常用组件	154
6.4.1 文本组件	154
6.4.2 特殊输入组件	155
6.4.3 按钮及其分类组件	156
6.4.4 列表组件	158
6.4.5 分组组件	161
6.4.6 信息反馈组件	162
6.4.7 表格显示组件	163
6.4.8 图形显示组件	164
6.5 小结	166
6.6 习题	166
第7章 代码书写标准	167
7.1 一般规则	167
7.2 Delphi 语言	168
7.2.1 括号	168
7.2.2 保留字	168
7.2.3 例程	168
7.2.4 变量	169
7.2.5 类型	170
7.2.6 构造类型	170
7.2.7 语句	171
7.2.8 异常处理	172
7.2.9 类	173
7.3 文件	174
7.3.1 项目文件	174
7.3.2 单元文件	174
7.4 窗体	175

7.5 组件	175
7.6 小结	177
7.7 习题	177
第8章 MDI 应用程序	178
8.1 创建 MDI 应用程序	178
8.1.1 理解 MDI 基础	178
8.1.2 主窗体	179
8.1.3 子窗体	184
8.1.4 补全代码	202
8.2 菜单	204
8.2.1 用 MDI 程序合并菜单	204
8.2.2 在菜单中列出打开的文档	205
8.3 小结	205
8.4 习题	205
第9章 多媒体应用程序	206
9.1 多媒体的概念	206
9.2 简易媒体播放器	207
9.3 MediaPlayer 组件	208
9.3.1 属性	208
9.3.2 事件	210
9.4 声音播放程序	210
9.5 视频播放程序	213
9.5.1 显示第一帧	213
9.5.2 Display 属性	214
9.5.3 DisplayRect 属性	214
9.6 CD 播放程序	215
9.6.1 编写 CD 播放器	215
9.6.2 更新 CD 播放器的信息	216
9.6.3 刷新 CD 播放器的方法	218
9.6.4 CD 播放器的源代码	219
9.7 小结	225
9.8 习题	225
第10章 文件处理	226
10.1 文件	226
10.1.1 打开与关闭文件	226
10.1.2 写入文本	227
10.1.3 读取文件	229
10.2 驱动器	229
10.2.1 驱动器列表	230
10.2.2 驱动器信息	231
10.3 目录	235
X	

10.3.1 获取 Windows 目录	235
10.3.2 获取系统目录	236
10.3.3 获取当前目录	236
10.3.4 在目录中查找文件	237
10.3.5 复制和删除目录树	239
10.4 小结	240
10.5 习题	240
第 11 章 系统信息与控制	241
11.1 获取系统信息	241
11.1.1 获取一般信息	242
11.1.2 获取进程信息	247
11.1.3 SysInfo 的源代码	249
11.2 控制系统行为	250
11.2.1 设置分辨率	251
11.2.2 关闭计算机	252
11.2.3 SysCtrl 的源代码	253
11.3 小结	254
11.4 习题	254
第 12 章 数据库开发	256
12.1 数据集	256
12.1.1 打开与关闭数据集	256
12.1.2 遍历数据集	257
12.1.3 编辑数据集	261
12.1.4 刷新数据集	264
12.1.5 数据集状态	264
12.1.6 数据集过滤	264
12.2 Table 组件	266
12.2.1 查找记录	266
12.2.2 主/细表	269
12.2.3 常用事件	270
12.2.4 创建数据表	271
12.3 其他数据集	272
12.3.1 Query 组件	272
12.3.2 文本文件数据库表	272
12.4 小结	276
12.5 习题	276
第 13 章 网络开发	277
13.1 套接字	277
13.1.1 阻塞与非阻塞连接	277
13.1.2 套接字连接	277
13.1.3 使用 TCP 组件	278

13.2 通信实例	279
13.2.1 CommServer	280
13.2.2 CommClient	287
13.3 发送邮件	292
13.4 小结	294
13.5 习题	294
参考文献	296

第 1 章 初识 Delphi

1.1 Delphi 简介

Delphi 是 Borland 公司推出的 Windows 应用程序开发工具,它具有方便、快捷、易学习、易使用等特点。

Delphi 的版本号已经由最初的 1.0 发展到如今的 Delphi 2005(9.0)。在这 9 个版本中,比较成功的版本是 7.0,并且该版本也比较适合初学者学习。(如无明确说明,后续部分中 Delphi 均指 Delphi 7.0。)

Delphi 7.0 安装并提供了一些新技术(例如 Windows XP 主题或 UDDI 等技术),变得更加强大,不过更重要的是这一版本还提供了一套可以被方便使用的第三方工具:RAVE 报告引擎、IntraWeb Web 应用程序开发技术和 ModelMaker 设计环境。

最后,Delphi 7.0 还向用户提供了首套用于编译 Pascal/Delphi 语言的 Borland 编译器(甚至也为老版本的 Delphi 提供了这一编译器),将不再对 Intel CPU 进行跟踪,而是转而跟踪 .NET CLI 平台。这一做法还为程序设计人员开创了一片新的天地。

可以说,Delphi 是一套非常好用的工具,同时它也是一套复杂的程序设计环境,该环境中包含着大量元素。本书将帮助读者掌握 Delphi 程序设计技巧,其中包括 Delphi 语言的使用、组件的使用、文件和驱动器的操作以及数据库功能的使用等,另外,还将说明在进行 Windows、Internet 程序设计时需要注意的元素。

Delphi 同许多其他的软件开发工具一样,按照不同层次和需要分成几个不同的版本。下面是当前的一些 Delphi 版本:

- 个人版(Personal)。主要针对的是新入门的 Delphi 用户和不常使用 Delphi 的程序员。该版本中不支持数据库编程,也不支持任何 Delphi 中的各种高级特性。
- 专业版(Professional)。主要针对专业开发人员。该版本包括了 Delphi 所有的基本特性,并增加了对数据库编程的支持、基本 Web 服务器的支持,以及一些外挂工具(包括 ModelMaker 和 IntraWeb)的支持。本书假设读者正在使用的至少是 Delphi 专业版。
- 企业版(Enterprise)和体系版(Architect)。它们的功能更强大,例如可以开发企业级应用等。本书将不对这两个版本的功能作深入探讨。

除了这些版本之外,还有一些可以定制 Delphi 环境的方法。本书中的屏幕图示,将尽量使用标准的用户界面。

1.2 集成开发环境(IDE)

Delphi 实际上是 Pascal 语言的一种版本,但它与传统的 Pascal 语言有很大差别。一个 Delphi 程序首先是应用程序框架(Frame),而这一框架正是应用程序的“骨架”。在框架上即使没有附着任何东西,仍可以运行。用户所要做的工作只是在框架中加入自己的程序。

默认的应用程序是一个空白的窗体(Form)或称为窗口,这个窗口具有 Windows 窗口的全部性质(比如具有标题栏、最大/小化按钮、关闭按钮,还可以移动、调整窗口大小等),只是还没有编写程序。因此,可以说应用程序框架通过提供所有应用程序所共有的东西,为用户应用程序的开发打下了良好的基础。

Delphi 已经为你做好了一切基础工作——程序框架就是一个已经完成的可运行应用程序,只是不处理任何事情。你所需要做的,只是在程序中加入你所需要功能的代码而已。而这一切的实现与一个界面友好、效率出众的 IDE 是分不开的。

1.2.1 IDE 概述

第一次启动 Delphi 时,我们会看到类似图 1-1 所示的 IDE。

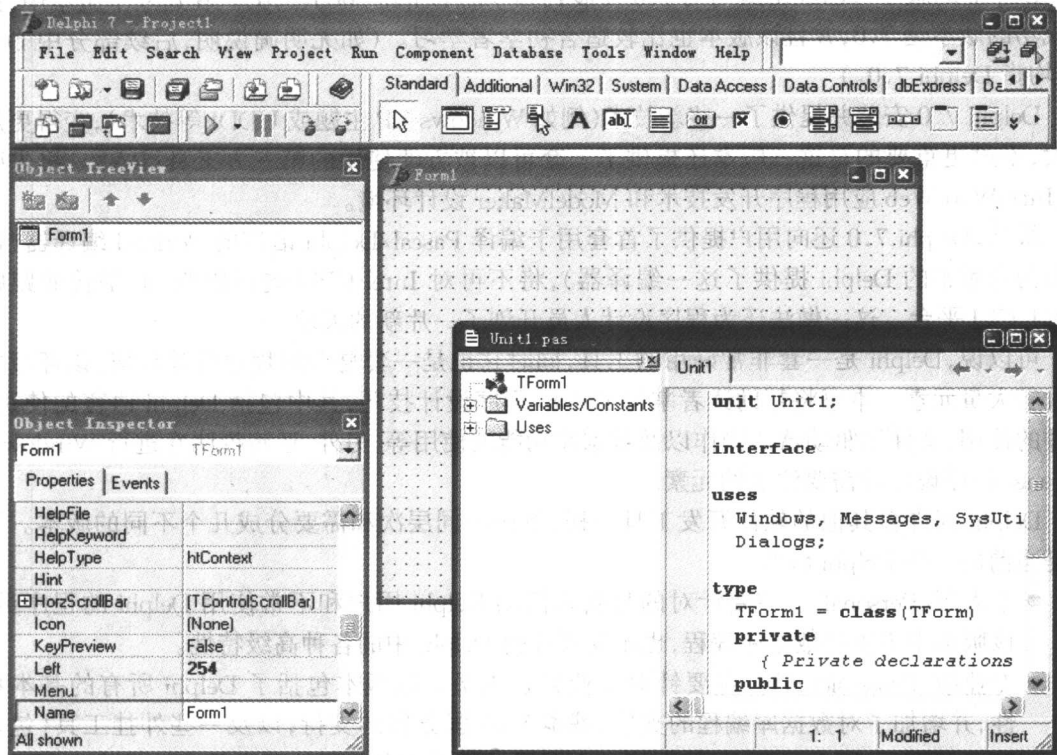


图 1-1 典型的 Delphi IDE

由图 1-1 可以看出,我们将要接触的 Delphi 的 IDE 基本分成以下几个部分:菜单栏、工具栏、组件面板、Object TreeView(对象树列表)、Object Inspector(对象查看器)、Form Designer(窗体设计器)和 Code Explorer(代码浏览编辑器)。另外当开发大型项目时,还会用到 Project Manager(项目管理器)。下面我们简要的介绍这几个部分。

1.2.2 菜单栏

尽管读者将来可能会通过使用快捷键和快捷菜单完成大部分的任务,但 Delphi 的主菜单栏仍然是初学者与 IDE 交互的重要方式。就我们当前的操作来说,不需要对菜单栏做太多的改变。

值得一提的是 Delphi 中的 Window 菜单。这个菜单列出了已经打开的窗口;其中的 Next Window 命令可以实现 IDE 内部不同窗口之间的切换。其对应的快捷键为〈Alt + End〉。

和标准的 Windows 程序一样,大部分菜单项目所对应的功能都可以通过工具栏上的按钮来调用。

1.2.3 工具栏

工具栏上的按钮提供了单击就可以访问菜单所提供的功能的特性。例如,打开一个文件或创建一个项目等。每个工具栏上的按钮都有一个提示。除去组件面板(严格的说,组件面板也是工具栏的一部分),IDE 中默认有五个工具栏,它们是:Debug(调试)、Desktop(桌面)、Standard(标准)、View(查看)和 Custom(自定义)。每个工具栏默认的位置可参考图 1-1。另外每个工具栏也可像图 1-2 那样单独形成一个独立窗口。

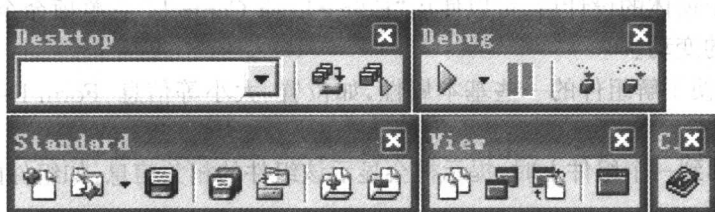


图 1-2 独立的工具栏

1.2.4 组件面板

组件面板是一个双层工具栏,它包含了 IDE 中安装的所有 VCL(可视化组件库)组件和 ActiveX 控件。这些组件分门别类放置在组件面板上,且数量非常多,足以满足用户绝大部分开发需要。

图 1-3 显示了组件面板中 Standard 选项卡中的组件,这些组件都是在编写程序时经常用到的。

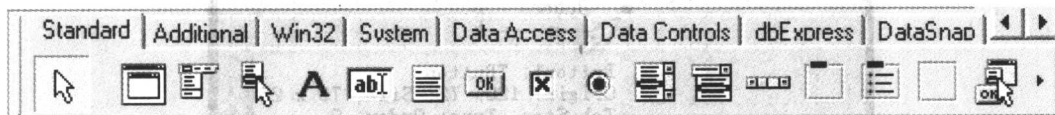


图 1-3 Standard 组件

1.2.5 Form Designer

如果想将组件面板中的组件放置到 Form Designer 中,可以直接双击该组件,或者先单击该组件并在 Form Designer 中再单击即可。

如图 1-4 所示,Form Designer 刚开始时是一个空白的窗体(Form)。可以把 Form Designer 看作是艺术家的画布,在这块画布上可以描绘出各种各样的 Windows 应用程序。应用程序的用户界面正是由窗体来实现的。

只要从组件面板上选择一个组件并把它放到窗体上,就能够实现与 Form Designer 的交互。可以用鼠标调整组件在窗体上的位置和大小,还可以用 Object Inspector 和 Code Explorer 来控制组件的外观和行为。

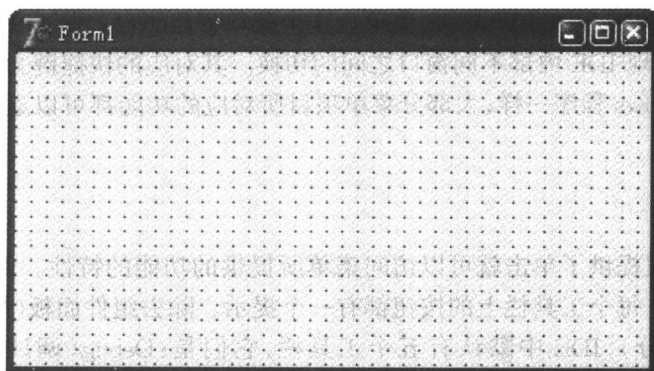


图 1-4 一个空白的 Form Designer

当完成了一个窗体的设计后,可以使用“Edit→Lock Controls”菜单项命令,以避免由于意外造成组件位置的变动。

另外,为了方便了解组件的一些基本属性,如位置、大小等信息,Form Designer 还提供了一些工具提示:

- 将鼠标移动到一个组件上时,提示框会显示该组件的相关信息,如组件的位置、大小、选项卡顺序等。
- 改变一个组件大小时,提示框会显示当前的大小。
- 移动一个组件时,提示框会说明当前的位置。

图 1-5 显示了第一种情况。

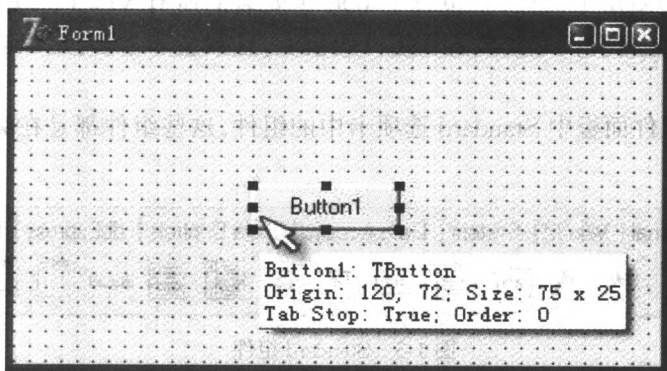


图 1-5 当鼠标移动到组件上时所显示的工具提示

1.2.6 Object Inspector

在设计程序时,经常要查看和修改放置在一个窗体中的组件的属性(Property)以及与该组件相关的事件(Event)。这时,我们可以使用 Object Inspector。

属性是一些数据,如高度、颜色、字体等,它们决定了组件在屏幕上的外观。事件则是一种消息处理机制,它能够捕捉某种情况的发生并作出反应,如鼠标单击和窗口重画就是两种典型的事件。

如图 1-6 所示,为了处理属性与事件,Object Inspector 中也相应包括 Properties 选项卡和 Events 选项卡。每个选项卡的左侧显示了属性(如 Caption)和事件(如 OnClick)列表,右侧则