

高等院校课程设计案例精编

赠送光盘  
中附有完  
整的案例  
源代码

# Delphi

## 课程设计

# 案例精编

孙福兆 雷志军 汪梅婷 余 健 编著

赠送光盘

- 学生信息管理系统 • 助学贷款信息管理系统 •
- 仓库管理系统 • 图书信息管理系统 •
- 贸易公司信息管理系统 • 组件使用与程序打包指南 •



清华大学出版社

TP311.56/426D

2008

## 高等院校课程设计方案精编

# Delphi 课程设计方案精编

孙福兆 雷志军 汪梅婷 余健 编著

清华大学出版社

北京

## 内 容 简 介

本书首先介绍了 Delphi 2006 的基本知识以及软件工程在实际中的运用,然后完整地介绍了 Delphi 在数据库开发方面的应用。全书共分 8 章,第 1 章和第 2 章介绍基本知识;其他各章均为实例,包括学生信息管理系统、助学贷款信息管理系统、仓库管理系统、图书信息管理系统和贸易公司信息管理系统。这些系统既能作为独立的系统运行,也可作为应用软件中的一部分,具有较强的实用性和可移植性。本书最后介绍了对第三方控件的使用以及程序的打包。

本书内容详实、语言简练、思路清晰、图文并茂、理论与实际设计相结合,适合作为高等院校计算机、自动化、机械、电子等相关专业学生课程设计指导书,也非常适合作为开发人员的参考用书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

Delphi 课程设计案例精编/孙福兆,雷志军,汪梅婷,余健编著.—北京:清华大学出版社,2008.7  
(高等院校课程设计案例精编)  
ISBN 978-7-302-17835-4

I. D… II. ①孙… ②雷… ③汪… ④余… III. 软件工具—程序设计—高等学校—教学参考资料  
IV. TP311.56

中国版本图书馆 CIP 数据核字(2008)第 084180 号

责任编辑:李春明 闫光龙

封面设计:山鹰工作室

版式设计:杨玉兰

责任校对:李凤茹

责任印制:何 芊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦 A 座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质 量 反 馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

印 装 者:北京宏伟双华印刷有限公司

经 销:全国新华书店

开 本:185×260 印 张:23.75 字 数:569 千字

版 次:2008 年 7 月第 1 版 印 次:2008 年 7 月第 1 次印刷

附光盘 1 张

印 数:1~4000

定 价:42.00 元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系  
调换。联系电话:(010)62770177 转 3103 产品编号:028032-01

# 前 言

课程设计是教学计划和环节中的重要组成部分,通过课程设计,学生能够得到系统的技能训练,巩固和加强所学的专业理论知识,其目的是培养学生的综合运行能力,使学生成为理论基础扎实和独立动手能力较强的专业人才。

Delphi 是著名的 Borland 公司开发的可视化软件开发工具。“真正的程序员用 C,聪明的程序员用 Delphi”,这句话是对 Delphi 最经典、最实在的描述。Delphi 被称为第四代编程语言,它具有简单、高效、功能强大的特点,利用 Delphi 进行数据库开发时,需要用到的相关控件和内容多,再加之学生在学到的编程理论知识往往比较模糊,只有通过真正动手编程,才有可能使学生对编程的理解从感性上升到理性。

本书首先介绍理论知识和编程需要的相关 API,使读者能知其然,然后通过五个完整的应用系统的制作开发,使读者能熟悉 Delphi 2006 开发步骤、方法和技巧,达到知其所以然的目的。本书提供的应用实例既具有针对性,且各自独立成章、分别进行讲述,具体内容如下。

案例一:介绍银行助学贷款信息管理系统,并使用第三方控件进行美化。

案例二:介绍用 SQL Server 2005 与 Delphi 2006 开发仓库管理系统,使用专门设计的图片达到美化界面的作用。

案例三:介绍与学生紧密联系的图书信息管理系统。

案例四:介绍与实际密切联系的贸易公司信息管理系统。

案例五:介绍了人事管理系统的开发重点,描述了对图像字段的操作。

案例六:介绍利用 InstallShield 创建安装程序。

本书由孙福兆、雷志军、汪梅婷和余健执笔,参加编写和程序开发的人员还有马秀萍、李双红、穆志维、伍建刚、陈伟、孙守凯、杨立平、吴宏彬、陈思成、孙景辉、方海刚、许钊等,在此一并向他们表示感谢。

由于编者水平有限,加上时间仓促,书中难免有一些不足之处,欢迎同行和读者批评指正。

编 者  
2008 年 3 月



# 目 录

|                                           |    |
|-------------------------------------------|----|
| 第1章 Delphi 2006 简介.....                   | 1  |
| 1.1 Delphi 简介.....                        | 1  |
| 1.1.1 Delphi 发展简史.....                    | 1  |
| 1.1.2 Borland Developer Studio<br>简介..... | 1  |
| 1.1.3 Delphi 编程特点.....                    | 2  |
| 1.2 Delphi 集成开发环境.....                    | 3  |
| 1.2.1 窗体.....                             | 5  |
| 1.2.2 工具窗口.....                           | 7  |
| 1.2.3 Model View.....                     | 13 |
| 1.2.4 改变环境设置.....                         | 13 |
| 1.3 Delphi 2006 新特性.....                  | 20 |
| 1.3.1 继承方式创建窗体.....                       | 20 |
| 1.3.2 强大的数据库支持.....                       | 21 |
| 1.3.3 代码自动完成功能.....                       | 22 |
| 1.3.4 改动指示.....                           | 22 |
| 1.3.5 实时模板.....                           | 22 |
| 1.3.6 包裹程序代码.....                         | 23 |
| 1.3.7 Debug.....                          | 23 |
| 1.3.8 Together for Delphi.....            | 24 |
| 1.3.9 ECO III.....                        | 24 |
| 1.4 新特性编程举例.....                          | 25 |
| 1.5 常用控件介绍.....                           | 30 |
| 1.5.1 Standard 控件.....                    | 30 |
| 1.5.2 Additional 控件.....                  | 33 |
| 1.5.3 Data Access 控件.....                 | 34 |
| 1.5.4 Data Controls 控件.....               | 34 |
| 1.5.5 dbExpress 控件.....                   | 35 |
| 1.5.6 BDE 控件.....                         | 37 |
| 1.5.7 dbGo 控件.....                        | 37 |
| 1.5.8 Servers 控件.....                     | 39 |
| 第2章 应用系统分析与设计.....                        | 41 |
| 2.1 应用系统设计的一般模式.....                      | 41 |
| 2.1.1 系统开发阶段的划分.....                      | 41 |

|                                |     |
|--------------------------------|-----|
| 2.1.2 软件开发方式.....              | 48  |
| 2.1.3 软件开发模型.....              | 48  |
| 2.1.4 软件开发方法.....              | 51  |
| 2.2 系统分析与设计方法.....             | 54  |
| 2.2.1 系统分析与设计过程中的<br>相关文档..... | 54  |
| 2.2.2 数据库设计.....               | 55  |
| 2.3 系统详细设计.....                | 58  |
| 2.3.1 系统功能分析.....              | 58  |
| 2.3.2 系统功能模块设计.....            | 59  |
| 2.3.3 数据流程图.....               | 62  |
| 2.3.4 数据库需求分析.....             | 64  |
| 2.3.5 数据库概念结构设计.....           | 64  |
| 2.3.6 数据库逻辑结构设计.....           | 66  |
| 2.3.7 数据库设计说明书.....            | 80  |
| 第3章 学生信息管理系统.....              | 82  |
| 3.1 课程设计目的和意义.....             | 82  |
| 3.2 系统总体设计.....                | 82  |
| 3.3 详细设计.....                  | 83  |
| 3.3.1 信息编辑模块设计.....            | 83  |
| 3.3.2 信息浏览模块设计.....            | 96  |
| 3.3.3 信息查询模块设计.....            | 99  |
| 3.4 系统运行结果.....                | 105 |
| 第4章 助学贷款信息管理系统.....            | 109 |
| 4.1 课程设计的目的和意义.....            | 109 |
| 4.2 需求分析.....                  | 109 |
| 4.2.1 功能需求.....                | 109 |
| 4.2.2 系统流程分析.....              | 110 |
| 4.3 总体设计.....                  | 110 |
| 4.3.1 系统层次模块图.....             | 110 |
| 4.3.2 数据库设计.....               | 111 |
| 4.3.3 数据库逻辑结构设计.....           | 112 |
| 4.3.4 数据表的创建.....              | 114 |

|                                    |     |                        |     |
|------------------------------------|-----|------------------------|-----|
| 4.4 主要技术介绍.....                    | 115 | 5.5.4 创建系统主界面 .....    | 204 |
| 4.4.1 ADO 控件简介 .....               | 115 | 5.5.5 供应商信息管理模块 .....  | 207 |
| 4.4.2 TADOConnection 控件<br>简介..... | 116 | 5.5.6 商品基本信息管理模块 ..... | 211 |
| 4.4.3 TADODataset 控件简介.....        | 117 | 5.5.7 商品入库管理模块 .....   | 215 |
| 4.4.4 TADOCommand 控件简介.....        | 119 | 5.5.8 商品出库管理模块 .....   | 220 |
| 4.4.5 TADO 其他控件 .....              | 120 | 5.5.9 商品查询模块 .....     | 224 |
| 4.4.6 ADO 创建过程.....                | 120 | 5.5.10 系统管理模块 .....    | 225 |
| 4.5 详细设计.....                      | 120 | 5.6 系统运行结果.....        | 230 |
| 4.5.1 启动界面设计.....                  | 121 | 第 6 章 图书信息管理系统.....    | 235 |
| 4.5.2 DataModule 的创建.....          | 122 | 6.1 课程设计的目的和意义.....    | 235 |
| 4.5.3 登录功能模块设计.....                | 124 | 6.2 需求分析.....          | 235 |
| 4.5.4 系统主窗口设计.....                 | 127 | 6.2.1 功能需求 .....       | 235 |
| 4.5.5 贷款学生信息管理模块.....              | 130 | 6.2.2 系统流程分析 .....     | 236 |
| 4.5.6 贷款信息管理模块.....                | 146 | 6.2.3 业务流程图 .....      | 237 |
| 4.5.7 系统管理模块.....                  | 157 | 6.3 总体设计.....          | 238 |
| 4.6 系统运行结果.....                    | 161 | 6.3.1 系统层次模块图 .....    | 238 |
| 第 5 章 仓库管理系统.....                  | 169 | 6.3.2 系统数据结构 .....     | 239 |
| 5.1 课程设计的目的和意义.....                | 169 | 6.3.3 数据库逻辑结构设计 .....  | 240 |
| 5.2 需求分析.....                      | 169 | 6.3.4 数据表的创建 .....     | 242 |
| 5.2.1 数据流图.....                    | 169 | 6.4 详细设计.....          | 244 |
| 5.2.2 系统主要功能.....                  | 171 | 6.4.1 主界面设计 .....      | 244 |
| 5.2.3 系统功能模块及说明.....               | 171 | 6.4.2 创建登录窗体 .....     | 250 |
| 5.3 总体设计.....                      | 171 | 6.4.3 图书管理模块的实现 .....  | 253 |
| 5.3.1 系统总体模块图.....                 | 171 | 6.4.4 书籍借阅模块的实现 .....  | 262 |
| 5.3.2 E-R 模型与实体模型 .....            | 172 | 6.4.5 读者开卡模块的实现 .....  | 270 |
| 5.3.3 数据库逻辑结构设计.....               | 173 | 6.4.6 查询管理模块 .....     | 271 |
| 5.3.4 数据表的创建.....                  | 176 | 6.4.7 系统管理模块的实现 .....  | 283 |
| 5.4 SQL Server 2005 数据库的应用.....    | 177 | 6.5 系统运行结果.....        | 295 |
| 5.4.1 数据库创建.....                   | 178 | 第 7 章 贸易公司信息管理系统 ..... | 301 |
| 5.4.2 表的创建.....                    | 182 | 7.1 课程设计的目的和意义.....    | 301 |
| 5.4.3 创建备份设备.....                  | 188 | 7.2 需求分析.....          | 301 |
| 5.4.4 数据库的备份与还原.....               | 189 | 7.2.1 系统主要功能 .....     | 301 |
| 5.4.5 配置 ODBC 数据源 .....            | 196 | 7.2.2 系统功能模块及说明 .....  | 301 |
| 5.5 详细设计.....                      | 199 | 7.3 总体设计.....          | 302 |
| 5.5.1 创建工程项目.....                  | 199 | 7.3.1 系统总体模块图 .....    | 302 |
| 5.5.2 创建数据库连接类.....                | 199 | 7.3.2 程序流程与实体模型 .....  | 303 |
| 5.5.3 用户登录模块.....                  | 202 | 7.3.3 数据库逻辑结构设计 .....  | 304 |

|        |              |     |
|--------|--------------|-----|
| 7.3.4  | 数据表的创建.....  | 306 |
| 7.4    | 详细设计.....    | 308 |
| 7.4.1  | 创建程序框架.....  | 308 |
| 7.4.2  | 创建数据连接类..... | 310 |
| 7.4.3  | 创建登录对话框..... | 314 |
| 7.4.4  | 顾客信息管理.....  | 315 |
| 7.4.5  | 商品信息管理.....  | 319 |
| 7.4.6  | 销售信息管理.....  | 322 |
| 7.4.7  | 供应商信息管理..... | 326 |
| 7.4.8  | 进货信息管理.....  | 330 |
| 7.4.9  | 相关查询模块.....  | 333 |
| 7.4.10 | 系统管理模块.....  | 344 |
| 7.5    | 系统运行结果.....  | 346 |

## 第 8 章 组件使用与打包指南.....354

|       |                             |     |
|-------|-----------------------------|-----|
| 8.1   | 第三方组件的应用.....               | 354 |
| 8.1.1 | 第三方组件简介 .....               | 354 |
| 8.1.2 | 第三方组件的安装 .....              | 355 |
| 8.1.3 | 第三方组件的使用 .....              | 361 |
| 8.2   | 使用 InstallShield 打包程序 ..... | 362 |
| 8.2.1 | 利用向导完成制作 .....              | 362 |
| 8.2.2 | 利用 IDE 修改安装工程.....          | 366 |
| 8.2.3 | 设置快捷方式 .....                | 367 |
| 8.2.4 | 设置注册表 .....                 | 368 |
| 8.2.5 | 设置 ODBC Resources.....      | 368 |
| 8.2.6 | 条件安装 .....                  | 369 |
| 8.2.7 | 生成与部署 .....                 | 370 |

# 第1章 Delphi 2006 简介

Delphi 2006 是 Borland 公司开发的面向对象的集成软件开发环境 Borland Developer Studio 2006 中的一员。Delphi 2006 不仅延续了 Delphi 的快速开发(Rapid Application Development, RAD)环境传统,还在许多领域扩展和增加了重要功能。

## 1.1 Delphi 简介

### 1.1.1 Delphi 发展简史

Delphi 是面向对象的 Pascal 语言,它的前身是 Pascal 语言。早期的 Pascal 语言,语法严格,除了支持常用的字符、数值等基本数据类型外,还支持子集、记录等结构数据类型。Pascal 语言的保留字和语法与英语很相似,所以易学易用,常常用作程序设计的入门语言。后来,随着面向对象技术的兴起,Pascal 中增加了对面向对象技术的支持,逐渐发展成为今天广为人知的 Delphi。

2002 年 8 月, Borland 公司推出了 Delphi 7 Studio,利用统一建模语言和模型驱动体系来加快系统开发。Delphi 7studio 全面支持 Microsoft 公司的 .NET 框架,提供 .NET 移植工具包,还对电子商务和 Web 服务框架进行了扩展。Delphi 7 Studio 共有 4 个版本:个人版、专业版、企业版和设计师版。

2003 年 11 月, Borland 公司又推出了 Borland Delphi 8 for the Microsoft .NET。开发者可以利用它创建基于 .NET 框架的应用。Delphi 8 共有 3 个版本:专业版、企业版和设计师版,其中设计师版的功能最为强大。

2004 年 11 月, Borland 公司发布更高版本的 Delphi 2005。Delphi 2005 支持多种编程语言,新增了代码编辑器的语法错误提示功能,支持命名空间,还引入了灵活的 For...In 语句,扩展了 For 循环。此外,还增加了对重构的支持等。

2005 年 12 月, Borland 公司为 Delphi 编程爱好者带来了惊喜, Delphi 的第 10 个版本——Delphi 2006 正式发行。除了继承 Delphi 2005 的原有功能外, Delphi 2006 融数据库开发技术、可视化程序设计技术和面向对象技术于一身,增加了很多新特性。例如,更新的数据库驱动程序、状态机、实时模板等。

### 1.1.2 Borland Developer Studio 简介

Borland Developer Studio 2006 家族的成员主要有三个: Delphi 2006、C++ Builder 2006 和 C# Builder 2006。

(1) Delphi 2006: 在面向对象的 Pascal 语言基础上发展而成,功能强大,提供了丰富的数据库开发控件,是快速开发数据库应用系统的理想选择。

(2) C++ Builder 2006: 采用 C++ 语言开发程序,支持面向对象程序设计。

(3) C# Builder 2006: 用 C# 语言开发应用系统,在网页开发方面有独到之处。



Borland Developer Studio 2006 的版本主要有三种：专业版、企业版和架构师版。尽管不同版本集成的新功能略有差别，每个版本都对所有的语言提供相同的支持。

### 1.1.3 Delphi 编程特点

#### 1. 可视化编程

Delphi 2006 的开发环境中处处体现着可视化的特点。从窗体界面设计中添加控件时的“画”控件，到建模时的可视化模型设计，都体现了可视化的特征，所见即所得。

#### 2. 采用事件驱动机制

事件是指发生在对象上的事情。Delphi 中的每个控件都有自身能够识别的事件。程序运行时，一旦某个事件被触发，系统将自动调用对应的事件过程处理代码，这种动作模式称为事件驱动。

#### 3. 语法检查严格

Delphi 的语法检查比较严格，因此书写语句时要遵循一定的规范，具体语句书写要求如下。

(1) 语句不区分大小写，开发人员可以按照自己的习惯书写，如 IF 和 If 在 Delphi 中表示相同的含义。

(2) 变量要先声明，后使用。Delphi 中的变量是显式声明的，这一点与 Microsoft Visual Basic 有明显的不同。变量没有声明而直接使用时，系统会报错，提示未声明的标识符(Undeclared Identifier)。

Delphi 中的变量命名规则如下。

- 必须以字母或下划线打头，后接字母、数字或下划线，中间不能有空格。
- 不能使用规定的保留字，尽量避免使用标准标识符。
- 变量名对大小写不敏感。
- 变量名的长度是任意的，通常只有前 255 个字符有效。

(3) 语句以分号结尾，可以写在一行或多行上，也可以一行书写多条语句。以下的书写格式都是正确的：

```
x:='Delphi';    y:='2006';    //一行写多条语句
x:=x+y;        //一行写一条语句
Edit1.Text := x +
'是 x 的值';    //一条语句写在多行
```

#### 4. 完全支持 Win32 程序设计和.NET 框架

Delphi 2006 的编程环境中既支持 Delphi for Win32 程序设计，又支持 Delphi for .NET，完全支持 Microsoft 公司的 .NET 框架。

#### 5. 提供丰富的开发控件，支持高效程序设计，而且编译速度很快

Delphi 2006 的控件有上百种，涉及界面设计、文档操作、数据库操作、网络开发、

统一建模等诸多方面。

## 6. 更新了数据库驱动程序

Delphi 2006 更新了数据库驱动程序, 与 Borland Data Providers for .NET 一起支持 SQL Server、InterBase、DB2、Oracle 等多种数据库。

## 7. 推出 ECO III

推出功能强大的 ECO III, 强化了模型驱动, 支持 UML 1.5 和 UML 2.0 的统一建模, 新增了对状态机的支持。

# 1.2 Delphi 集成开发环境

Delphi 2006 提供了可视化的编程环境, 利用 Delphi 可以快速开发出基于 Windows 环境的各类应用程序。由于 Delphi 2006 提供了强大的数据库支持, 使用方便、灵活, 因此是快速开发各种数据库应用系统的理想选择。

启动 Delphi 2006 后进入集成开发环境, 如图 1-1 所示。

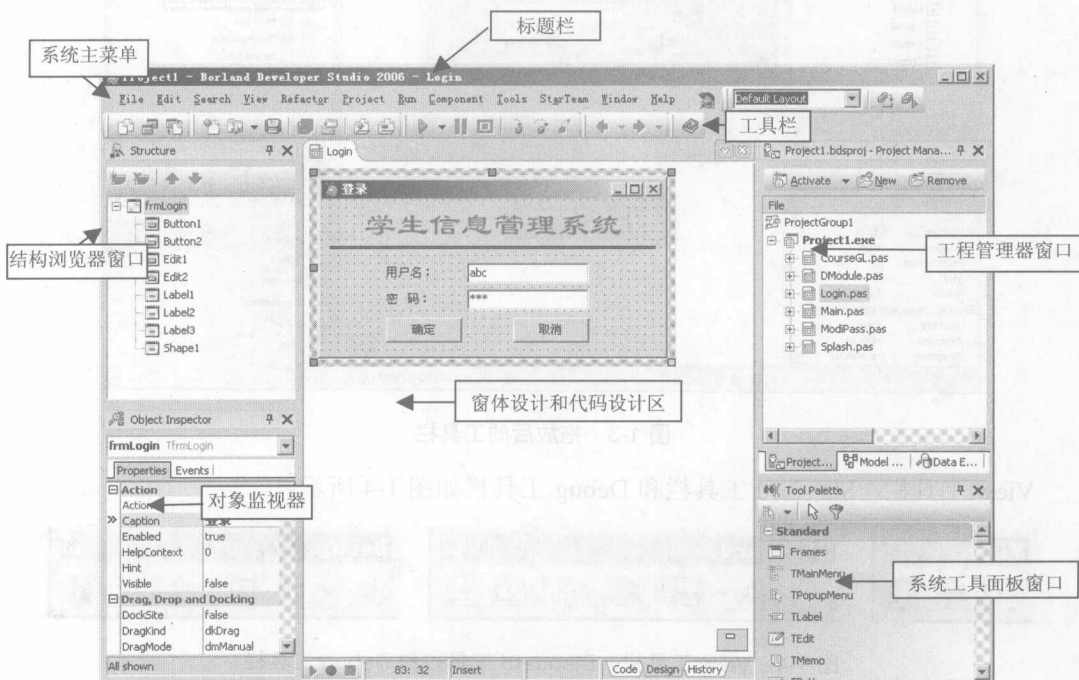


图 1-1 Delphi 2006 集成开发环境

Delphi 2006 的标题栏位于窗体的顶部, 工程项目打开后, 当前工程项目的名称显示在图标之后。紧接着是开发语言, 之后显示的是当前的程序文件名。在运行程序时, 标题栏上还将出现 “[Running]” 字样。

标题栏的下方是系统的主菜单，再向下是工具栏。Delphi 2006 的菜单栏和工具栏如图 1-2 所示。

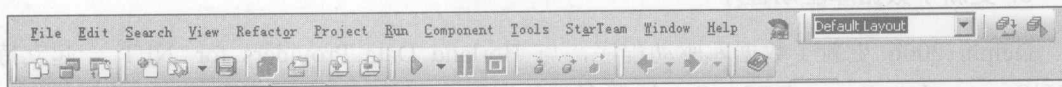


图 1-2 Delphi 2006 的菜单栏和工具栏

主菜单里包含集成环境的文件操作、程序代码编辑、调试、编译、运行等命令，通过鼠标单击选择。主菜单的选择也可以通过键盘操作完成，具体作法是按下组合键 Alt+带下划线的字母。例如，同时按下 Alt+F 可以打开 File 菜单。

工具栏位于主菜单的下方，Delphi 的工具栏可以自由拖放，可以根据用户的需要随时增加或关闭。图 1-3 展示的是拖放后工具栏在界面的位置。

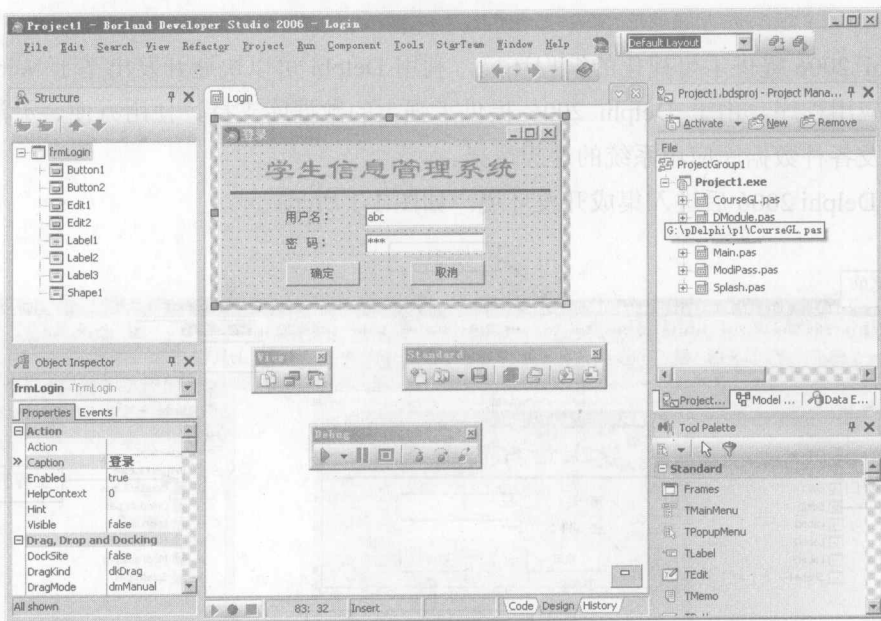


图 1-3 拖放后的工具栏

View 工具栏、Standard 工具栏和 Debug 工具栏如图 1-4 所示。

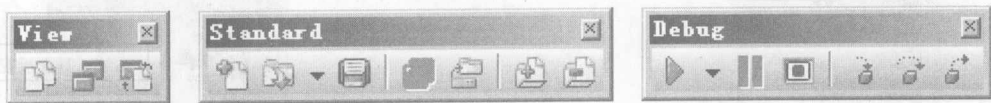


图 1-4 View 工具栏、Standard 工具栏和 Debug 工具栏

(1) View 工具栏的按钮功能从左至右依次为查看单元(Unit)，查看窗体(Form)界面，窗体界面与代码窗口切换。

(2) Standard 工具栏的按钮功能具体如下。

- 按钮功能是添加新项，单击此按钮将出现图 1-5 所示的窗口。先在左侧的树状列表中选择类别，后在右侧的列表视图选择要添加的窗体、对话框等。

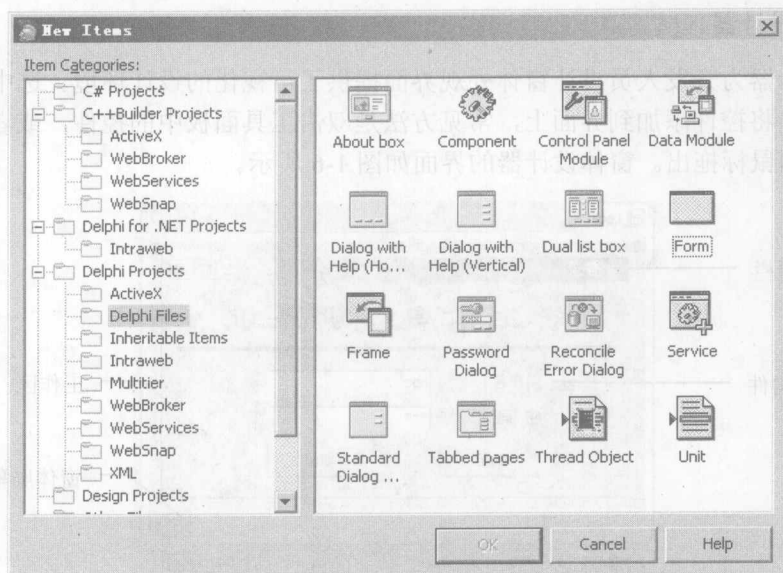


图 1-5 New Items 窗口

- 按钮功能是打开，单击按钮右侧的箭头，从弹出的下拉菜单中选择要打开的工程项目或文件。
- 按钮功能是保存当前窗体或模块，可以通过按下组合键 **Ctrl+S** 实现存盘。
- 按钮功能是保存工程，也可以通过按下组合键 **Shift+Ctrl+S** 实现。
- 按钮功能是打开工程，也可以通过按下组合键 **Ctrl+F11** 实现。
- 按钮功能是向工程中添加文件，也可以通过按下组合键是 **Shift+F11** 实现。
- 按钮功能是从工程中删除文件，单击后从弹出的对话框中选择要删除的窗体，然后单击【OK】按钮，此时被选定文件就从工程中删除。

(3) Debug 工具栏提供了程序代码运行和调试的常用按钮，按钮功能具体如下。

- 按钮功能是运行程序，也可以通过按下 **F9** 键执行。
- 按钮功能是暂停执行。
- 按钮功能是重置程序，也可以通过按下组合键 **Ctrl+F2** 实现。
- 按钮功能是跟踪运行，也可以通过按下 **F9** 键执行。
- 按钮功能是结束跟踪运行，也可以通过按下 **F8** 键实现。
- 按钮功能是跟踪运行到返回为止，也可以通过按下 **Shift+F8** 键执行。

### 1.2.1 窗体

窗体是应用程序设计期间的操作界面，是运行时程序与用户进行信息交流的桥梁。设计状态下，编程人员通过添加控件，设置窗体属性的方法来美化窗体的外观。编程时，为窗体和窗体上的控件添加代码后，完成特定的功能。

窗体由标题栏和工作区两部分组成：标题栏显示窗体的标题和用于实现最大化、最小化、还原和关闭功能的控制按钮；工作区可以容纳菜单、工具栏和其他控件。

## 1. 窗体设计器

窗体设计器为开发人员设计窗体外观界面提供了可视化的设计环境。选中窗体后，有多种方法可以将控件添加到界面上，常见方法是双击工具面板中的控件，或者先选中控件后在窗体上用鼠标拖出。窗体设计器的界面如图 1-6 所示。

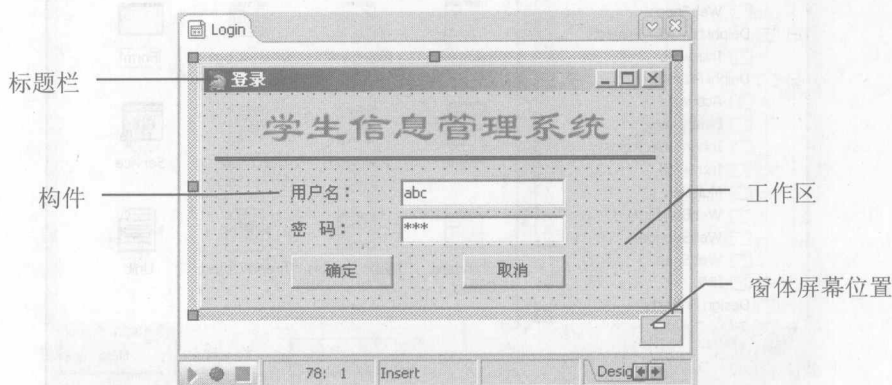


图 1-6 窗体设计器

注意，图 1-6 所示窗体设计器的右下角有一个灰色的矩形，内置一个很小的矩形框。灰色矩形代表运行时的屏幕，而内置的矩形框代表当前的窗体。二者的相对位置展示了运行时窗体和屏幕的位置关系。

需要改变窗体与屏幕的位置关系时，用鼠标选中代表当前窗体的小矩形，之后按下鼠标左键进行拖动。随时鼠标位置的改变，内部的小矩形的位置也随之发生变化，小矩形所在的位置就是程序运行时窗体相对于桌面的位置。选定位置之后，松开左键。通过这种可视化的方式，无需运行程序就可以提前知道窗体运行时的位置，而且调整起来更直观、更快捷。

## 2. 窗体的属性

属性是描述对象特征的参数，名称、颜色等都是常用属性。通过改变属性的值来决定对象的外观或特性。窗体的常用属性如表 1-1 所示。

表 1-1 窗体的常用属性

| 属性名     | 功 能  | 属性名          | 功 能                 |
|---------|------|--------------|---------------------|
| Name    | 名称   | BoarderStyle | 边框样式                |
| Caption | 窗体标题 | FormStyle    | 风格：MDI 父/子窗体、正常、恒置上 |
| Height  | 高度   | Icon         | 图标                  |
| Width   | 宽度   | Visible      | 是否可见                |
| Font    | 字体设置 | WindowState  | 初始状态：最大/小化、正常大小     |

## 3. 制作展示屏

程序运行时第一个出现的瞬间自动消失的欢迎界面称为展示屏，这是一个没有标题栏的窗体，主要展示应用系统的名称、版本、开发者以及版权声明等信息。



制作展示屏的步骤并不复杂,通过简单的属性设置,就可以把普通窗体变成一个展示屏,具体操作如下。

(1) 添加窗体,将窗体的 `BorderStyle` 属性值设置为 `bsNone`。

(2) 添加名为 `Timer1` 的 `TTimer` 控件,将 `Enabled` 属性值设置为 `True`, `Interval` 属性值设置为 1 000。

(3) 双击 `Timer1`,输入 `Timer1` 的 `OnTimer` 事件代码,实现卸载展示屏,显示主界面的功能。

一个简单的展示屏就这样完成了。

## 1.2.2 工具窗口

### 1. 代码编辑器

代码编辑器是用户编写程序代码和添加事件处理过程的地方,每当用户创建一个新窗体时,代码编辑器就自动生成一个新单元,其内容与窗体的程序文件(\*.pas)对应。在设计状态下,双击窗体就可以打开代码编辑器(Code),也可以通过双击工程管理器中程序文件的名称(如 `Login.pas`)打开。代码编辑器如图 1-7 所示。

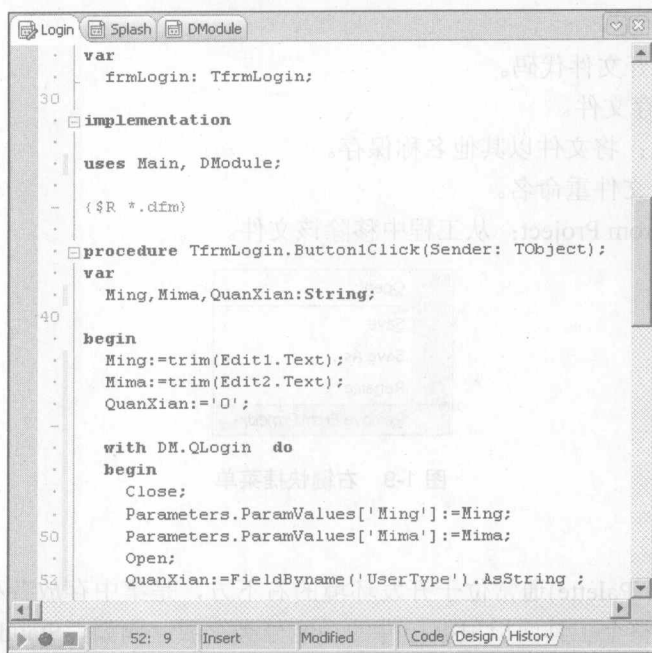


图 1-7 代码编辑器

代码编辑器的彩色文法功能支持以彩色的方式显示程序代码。采用默认设置时,程序保留字以深蓝色显示,注释语句以绿色显示,而普通代码则以正常显示。

当用户同时打开多个窗体的代码时,代码编辑器以选项卡的形式依次排列。通过顶部选项卡的标题(如 `Login`)区分代码所属的程序文件。单击选项卡标题就可以完成窗体代码的切换。

## 2. 工程管理器

工程管理器(Project Manager)窗口采用树状列表的形式显示当前工程项目中的各类文件,完成文件的打开、保存、重命名、添加和删除等功能。工程管理器的界面如图 1-8 所示。

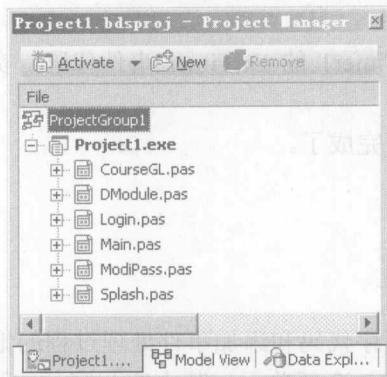


图 1-8 工程管理器窗口

右击工程管理器中的文件时,会弹出图 1-9 所示的快捷菜单。各个菜单项的功能如下。

- Open: 打开文件代码。
- Save: 保存文件。
- Save As...: 将文件以其他名称保存。
- Rename: 文件重命名。
- Remove From Project: 从工程中移除该文件。

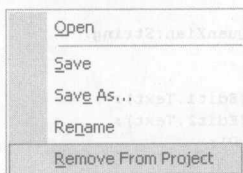


图 1-9 右键快捷菜单

## 3. 工具面板

工具面板(Tool Palette)通常位于开发环境的右下方,是集中存放各种控件的区域。为了方便用户使用,整个工具面板根据控件的功能分类存放。例如,Standard 页存放设计界面常用的标准控件,如标签、命令按钮、编辑框、单选按钮等;Dialogs 页存放各种常用的系统对话框,如颜色对话框、字体对话框、打开对话框等。

控件种类很多,不能一次显示所有控件,所以 Delphi 2006 提供了展开和折叠控件页的功能。单击每个分类名前的方框,出现减号“-”时控件页展开,出现加号“+”时控件页折叠。

工具面板的界面如图 1-10 所示,图中显示的是 Standard 标准控件页,当前处于展开状态。拖动窗口右侧的垂直滚动条,可以看到下方被遮挡起来的其他控件。

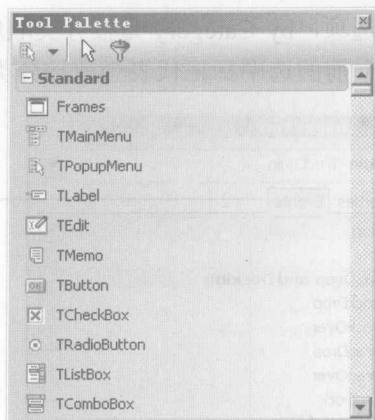


图 1-10 工具面板的界面

#### 4. 对象监视器

对象监视器(Object Inspector)通常位于 Delphi 编程环境的左侧,用于编辑和修改对象属性,以及查看和修改对象事件名称。

对象监视器的上部是一个下拉列表框,列表中的每个条目是一个对象。当下拉列表中被选中的对象改变时,对象监视器下方选项卡上的显示内容也随之改变。对象监视器共有两个选项卡,一个是 Properties(属性选项卡),另一个是 Events(事件选项卡)。Properties 选项卡用于查看和设置对象的属性值,而事件选项卡用来查看事件和修改事件代码对应的过程名。

对象监视器的 Properties 选项卡如图 1-11 所示,Event 选项卡如图 1-12 所示。

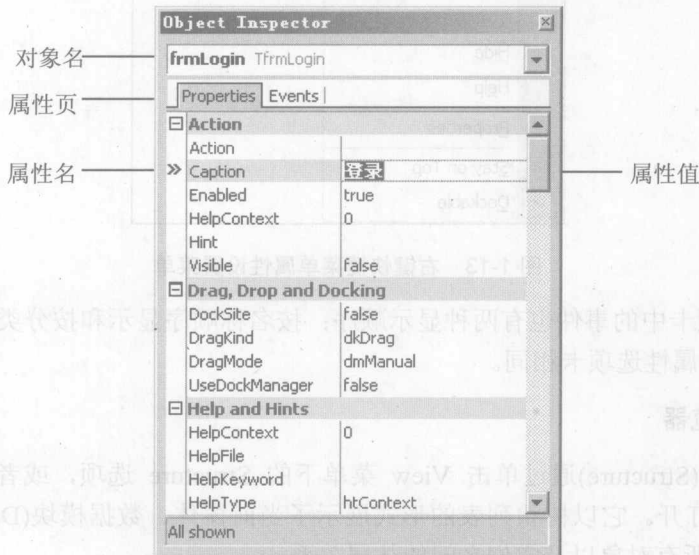


图 1-11 对象监视器的 Properties 选项卡

Properties 选项卡中的属性有两种显示顺序:按名称顺序显示和按分类顺序显示。设置属性显示顺序时,先右键单击对象编辑器,弹出图 1-13 所示的菜单之后,选中

Arrange。Arrange 下有两个子选项，by Category 是按分类顺序显示，by Name 是按名称顺序显示。图 1-13 中 by Category 前的选中标记代表当前按分类顺序显示。

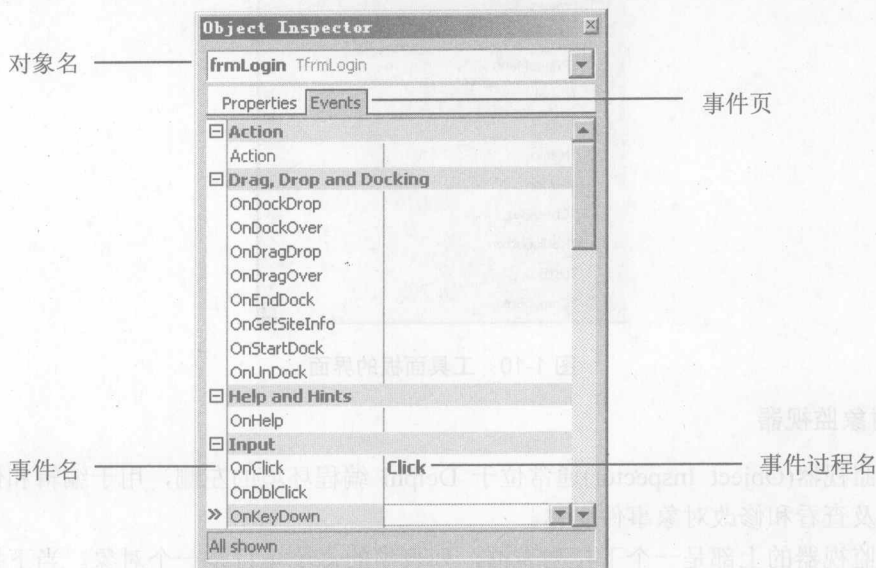


图 1-12 对象监视器的 Events 选项卡

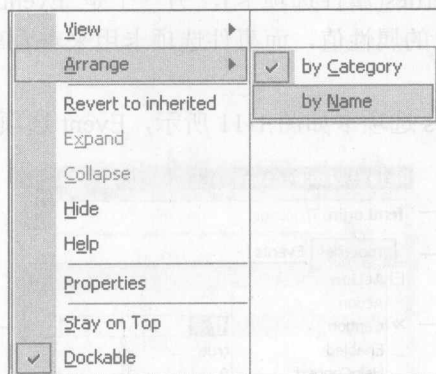


图 1-13 右键快捷菜单属性设置菜单

Events 选项卡中的事件也有两种显示顺序：按名称顺序显示和按分类顺序显示。设置排序方式步骤与属性选项卡相同。

## 5. 结构浏览器

结构浏览器(Structure)通过单击 View 菜单下的 Structure 选项，或者同时按下组合键 Shift+Alt+F11 打开。它以树状列表的形式展示了当前窗体、数据模块(Data Module)或框架(Frames)中的所有对象以及它们之间的从属关系。

结构浏览器中展示的对象包括可视对象和不可视对象。可视对象在设计状态和运行状态都可见，而不可视对象只在设计状态可见，在程序运行时不可见。例如，编辑框(TEdit)是可视的，而计时器(TTimer)是不可视的。结构浏览器的界面如图 1-14 所示。