

专家
门诊
系列丛书



求是科技 张增强 李鲲鹏 编著

专家门诊—— Delphi 开发答疑

300问

人民邮电出版社
POSTS & TELECOM PRESS

专家
门诊
系列丛书

■ 求是科技 张增强 李鲲鹏 编著

专家门诊—— **Delphi** 开发答疑

300问

人民邮电出版社

前言

本书精选了 300 多个实用性极强的问题和需求，旨在帮助读者进行 Delphi 开发中解决诸多实际问题，获得大量的功能实现方法。

本书内容涉及到 Windows 编程的各个方面，包括 IDE 的使用、Object Pascal、初级控件的使用、界面设计、响应处理、高级控件应用、图形图像、多媒体应用、文件处理、Windows 外壳处理、程序控制、高级应用、数据库应用、网络编程。

本书以专家解答的形式进行讲解，内容涵盖全面，按照实际开发中的功能目的进行分类，是广大 Delphi 开发和使用人员解决实际问题、积累编程经验的得力助手。本书重点突出程序设计实用性，同时也安排了一部分帮助读者理解重点、难懂概念的内容。

每个问答或者需求都根据技术难度的不同加以标识，并给出在实际开发中的人气指数（经常涉及频繁程度）。

每个问答的内容包括：

- 问题简述——用简洁的标题描述问题和需求。
- 问题详述——将问题和需求讲述清楚（背景、情况、需求、状况等）。
- 专家解答——给出问题的解决办法和满足需求的解决方案。并做适当扩展，使读者获得更多知识。

对于编码性技术问题，文档中只给出关键代码，完整程序的放在光盘上。对于方案性和方法性的解答，不在文档中给出详细的代码，只是侧重语言描述清楚，完整的程序代码也放到光盘里，但讲解时会有效地结合光盘中代码。

本书由求是科技策划，张增强，李鲲程编写，特别感谢张宏林在技术上的支持。此外，以下人员也参与了本书的资料搜集和写作工作，他们是刘冰玉、杨柯岚、王保东、杨珏、赵昊彤、程凡、程卫峰、范桂山、莫微、宋征、王保东、吴频、邢庆子、徐旻、闫华文、袁亚玓、岳进、钟明、黄成昆、王远、吴晓超、肖永顺、钱力鹏、骆小来、李岩松、王雅琴、王芳、王国红等。以上人员对本书的顺利完成付出了辛勤的汗水和心血，在此一并表示感谢。

由于时间仓促，加之水平有限，书中的缺点和不足之处在所难免，敬请读者批评指正，本书责任编辑的联系方法：zhanglike@ptpress.com.cn，欢迎来信交流。读者也可以登录求是科技的网站 <http://www.cs-book.com> 进行交流。

编者

2003 年 6 月

目 录

第 1 章 IDE 应用	1
如何向项目中添加和删除单元	2
如何实现控件面板的自动滚动	3
如何控制代码的执行位置	3
快速编译项目和 PAS 文件	3
Delphi 源文件中包括哪些文件, 各自的作用	
如何	5
如何保存自己的编程环境	8
如何将做好的界面添加到模板中	8
Delphi 中包含哪些常用的快捷键	10
第 2 章 编程基础	11
如何初始化数组和记录	12
如何将动态数组中的内容写入文件	12
如何创建和使用自定义对象	13
如何计算 x 的 y 次方	18
第 3 章 字符串操作	19
如何设定和取得字符串的长度	20
如何将两个字符串连接起来	20
如何取得字符串中指定位置的字符	21
如何将字符串中的字符依次写入数组	21
如何对字符串按照一定的规则进行分割	21
如何将字符串改为首字大写	22
如何替换字符串中的内容	23
如何去掉字符串中的空格	24
如何统计字符串中中英文的字数	24
第 4 章 文本操作控件	26
如何将 Label 控件设置为透明状态	27
如何更改标签页的高度	27
如何隐藏 Memo 控件中的光标	28
如何实现 Memo 控件中文字查找功能	28
如何将指定文件的路径拖动到 Memo 控件中	
.....	29
如何得到 Memo 控件中的字符总数	31
控制 Memo 控件的滚动条	31

如何设定 Memo 控件的边框	32
使 Edit、Memo 原有的右键功能失效	32
如何限制 Edit 控件只输入数字和小数点	33
如何让一个 Edit 内的字符显示到它的右边	
.....	33
如何同时清除多个 Edit 控件中的内容	35
如何改变 RichEdit 控件中文本的样式	35
如何控制 RichEdit 中的自动换行	36
如何取得密码框中的内容	36
第 5 章 列表控制控件	38
如何更改 StringGrid 控件的颜色	39
在 StringGrid 中实现文字换行	40
如何制作属性列表框	41
使用 ListView 控件显示指定目录中的文件名	
.....	41
如何建立树形列表	42
如何删除 TreeView 控件中的列选项	44
使用 TreeView 控件显示指定目录中的内容	
.....	44
如何快速删除 TreeView 控件中的内容	45
如何将 ListBox 中的项目移动到另一个中	46
响应 ListBox 控件的滚动条事件	46
如何为 ListBox 控件添加水平滚动条	48
实现 ComboBox 控件的自动跟踪功能	48
使用代码控制 ComboBox 控件的下拉效果	
.....	49
智能查找列表项中的内容	50
如何拖放列表项中的对象	52
第 6 章 按钮控件	54
如何制作椭圆按钮	55
如何制作动画按钮	56
如何在按钮中显示两行文字	57
如何使按钮中的文字竖直显示	57
取得按钮被按下的时间长度	58
使多个按钮共享一个事件	59

第 7 章 时间显示与控制	61
如何制作小日历	62
如何取得系统日期和系统时间	64
如何取得毫秒级的时间段	64
第 8 章 一般窗口及界面设计	67
如何制作进程条	68
如何分割窗体	68
如何制作带图片的下拉条	69
如何制作颜色下拉条	71
如何快速调用各种常用对话框	71
如何在对话框中预览图片	73
如何响应窗体标题栏事件	74
如何把 popupmenu 触发事件改为左键触发	75
如何截获窗体的关闭信息	75
如何向状态条中添加图像	76
如何实现透明窗体	77
如何实现不规则窗体	78
如何实现任意位置可拖动的窗体	80
如何实现文字窗体	81
如何实现颜色渐变窗体	81
如何将图片作为窗体的背景	82
如何控制窗体的滚动条	83
如何隐藏标题栏	84
如何实现翻转窗体	84
如何限制窗体的移动	85
如何限制窗体最大化时的大小	86
如何隐藏窗体最小化时状态栏的图标	87
如何在程序的标题栏绘制文字	88
如何判断窗体是否被移动	88
如何使窗口始终显示到最前端	89
如何使用 PageDown 键实现窗体的滚动	89
如何制作倾斜文字	90
如何实现颜色渐变的进度条	92
如何使窗体永远最小化	93
如何向 DrawGrid 中添加图像	94
给窗体边框加颜色	94
如何在标题栏上添加按钮	95

如何创建及排列多文档窗体	97
如何控制 MDI 子窗体最大化时的大小	99
如何以动态效果显示窗体	100
如何实现控件的阴影	103
第 9 章 菜单操作	105
如何动态生成菜单项	106
如何制作位图菜单	108
如何修改系统菜单	110
如何实现菜单项分组	112
如何设置菜单的加速键和热键	113
如何设置级联菜单	114
如何控制菜单的有效和无效	114
如何控制菜单的复选标记	114
如何实现单选菜单	116
第 10 章 鼠标和键盘响应	118
什么是消息	119
如何随时响应用户的输入	119
如何控制 Power 键	120
如何检测用户按下的是否为 Enter	121
如何检测 Shift 、 Alt 和 Ctrl 键是否被按下	121
如何模拟按键	122
如何屏蔽系统功能键	122
如何记录屏幕操作	123
如何在按下键盘时弹出快捷菜单	125
使用方向键在控件中间切换	126
锁定任务栏	127
如何截获 Edit 控件中的 Tab 按键	127
如何按下 Esc 键退出程序	128
如何热键随时激活程序	128
如何设定光标的位置	129
制作光标	130
如何更改系统的鼠标光标	131
如何实现双击和单击操作	132
如何在一段时间后隐藏鼠标	133
如何限制鼠标位置	134
如何截获鼠标移开事件	135
如何截获按键的 OnClick 消息	135

第 11 章 图形图像	137
如何绘制各种常用图形	138
如何利用画布绘制各种图形	139
如何将 BMP 文件转换为图标文件	139
如何将图像翻转显示	140
如何复制图片的一部分	140
如何缩放图片	141
如何在 Image 控件中添加文字	141
如何抓取屏幕	142
如何把彩色图片转为灰度	144
如何取得某点的颜色值	145
如何将图片转换为 256 色	145
将文本内容保存为位图	146
第 12 章 多媒体	148
什么是 OpenGL, 它有什么功能	149
使用 OpenGL 实现镜面反射效果	149
如何对材质进行填充	152
如何制作视频播放器	154
制作 CD 播放器	155
如何播放 RM 文件	157
如何播放 Flash 文件	159
如何全屏播放电影	160
如何将 rc 文件转换为 Res 文件	160
如何禁止光驱自动播放光盘中的内容	161
如何判断光驱中是否有 CD	161
如何弹出和关闭光驱	163
如何获取 Audio-CD 的序列号	163
如何检测声卡是否存在	166
如何调用声音属性设置窗口	167
如何调用简单的声音	167
第 13 章 打印	168
如何获取打印机信息	169
取得可执行文件的图标	169
如何打印文本和位图	170
如何获得打印机的驱动和端口信息	171
如何更改默认的打印机	173
如何得到打印的页边距	174
如何打印 RTF 文件	175

如何改变纸张的打印大小	175
如何设置打印方向	176
第 14 章 硬件与操作系统	178
如何获取和修改计算机名称	179
如何修改显示器的分辨率	180
如何锁定计算机	182
如何获取 Windows 系统的版本信息	183
如何获得操作系统的序列号	185
如何获取 CPU 信息	185
如何获得内存的使用情况	186
检查驱动器是否准备好	187
如何列举系统可用的字体	188
如何隐藏任务栏	188
如何更改桌面	189
如何获得任务栏以外的桌面大小	192
如何隐藏桌面图标	192
如何使系统背景变灰	193
第 15 章 程序控制	194
什么是线程和进程	195
如何制作多线程程序	195
如何设定关联文件	198
如何将文件添加到历史记录中	200
如何禁止窗体右上角的按钮	201
如何使程序只执行一个实例	202
如何隐藏程序	203
如何为程序创建快捷方式	204
如何制作托盘程序	206
如何在程序中改变窗口的状态	209
如何查阅可视窗口的标题	209
如何保存程序的运行状态	210
如何将声音文件嵌入于可执行文件中	212
如何启动屏幕保护程序	213
如何响应窗体最大化事件	214
如何转让程序的控制权	214
如何为程序建立注册机制	215
如何控制开始菜单	216
如何关闭外部程序	217
如何使其他应用程序作为当前窗口	218

如何启动计算器	218
如何启动资源管理器	218
如何启动 IE 浏览器, 并连接到指定的网址	219
如何启动控制面板中的程序	220
如何使用剪贴板	221
如何实现程序间传递数据	222
利用内存映射实现数据交换	224
如何查看 IE 浏览器是否在运行	226

第 16 章 磁盘文件.....227

如何制作资源管理器	228
如何使用 OpenFileDialog 控件同时打开多个文件	229
如何得到 FileListBox 中选中的文件名	230
如何更改目录的名称	230
如何查找指定文件夹下的文件	231
如何将长文件名转为短文件名	232
如何更改文件的扩展名	233
如何实现文件夹的选定	234
如何创建和删除文件夹	235
如何创建深层目录	237
如何清空回收站	237
如何将文件删除到回收站中	238
如何获取文件属性	239
如何获得同相应扩展文件名关联的应用程序	239
如何取得系统目录	241
如何得到当前文件所在的目录	241
如何查看文件是否存在	242
如何创建和使用临时文件	242
如何读取文本文件的信息	242
如何复制文件	243
如何将文件缓冲区的内容写入硬盘	243
使用流文件读取数据	244
如何打开 Chm 帮助文件	244
如何修改文件的创建时间和修改时间	245
如何取得所有的光驱盘符	246
如何查看驱动器类型	247

如何获取驱动器容量	248
-----------------	-----

第 17 章 数据库操作.....249

如何创建一个 Paradox 数据库	250
如何进行数据库基本操作	250
如何以独占的方式打开数据库	253
如何获得 BDE 的系统信息	253
如何添加、修改和删除记录	254
如何判断指针记录是否改变	255
如何设定数据的显示范围	256
如何清空表中的内容	257
如何在 Query 控件中使用 SQL 语句	257
如何利用 Table 的过滤机制实现动态查询和统计	258
取得数据库中所有表的名称	259
如何判断表是否存在	260
如何取得表中的字段名称	260
如何设置当前索引	261
如何取得指定表的记录数目	261
如何判断是否读到了记录的末尾	262
如何建立临时表并将其内容保存到永久表中	262
如何创建一个和原始表结构相同的新表	263
如何将一个字段中的内容导入到 ListBox 中	264
如何将数据库中的内容添加到 OutLine 控件中	264
如何将数据库中的数据以图表的方式显示出来	265
如何向图表中插入背景图片	267
如何在 DBGrid 控件中显示图片	267
如何取消 DBGrid 控件自动添加空记录	269
如何修改 DBGrid 控件中指定行的颜色	270
如何通过单击 DBGrid 标题进行排序	271
如何转换数据库	272
如何确定并提示数据库正在使用	273
如何判断重复的值并进行提示	273
如何存取数据库中的图像	275
动态创建 ODBC 数据源	278

如何在数据库程序中使用多线程技术.....	280	如何向 IE 的工具栏中添加按钮.....	327
如何创建数据库登录程序.....	283	如何设置程序自动启动.....	328
如何使用数据模块.....	285	为程序创建快捷方式.....	329
如何使用书签.....	286	如何限定程序的使用次数.....	331
如何使用 ADO 打开带有密码的数据库 ..	287	第 20 章 其他.....	333
如何建立 MTS 服务器.....	288	如何取得程序中使用的控件类型.....	334
如何建立 MTS 访问客户端	291	如何取得汉字的第一个拼音字母.....	334
第 18 章 网络与通信	293		
获取本机并修改 IP 地址.....	294		
如何检测计算机是否联网.....	295		
如何知道系统有几个端口.....	295		
如何获取网络中所有的工作组	296		
如何取得计算机所在的工作组的名称.....	298		
如何更改 DNS 地址	299		
如何根据 IP 地址得到计算机的主机名称	300		
如何实现 Ping 操作.....	301		
如何映射网络驱动器.....	303		
如何检查网络上的计算机是否存在	304		
如何取得目前网络连接状态	305		
如何浏览打印机连接端口.....	306		
如何中断网络驱动器的连接	306		
如何取得网络上的用户名.....	306		
如何取得网络资源的名称.....	307		
如何取得 WNet API 的错误消息	307		
取得 Modem 的状态.....	308		
如何检测拨号状态.....	309		
如何编写 WWW 浏览器	311		
如何创建动态网页.....	313		
如何显示 XML 文件.....	316		
什么是远程服务	318		
如何建立和挂起远程连接.....	318		
如何建立远程数据库服务器	319		
如何与远程数据库服务器通信	321		
如何编写聊天室程序	321		
第 19 章 注册表操作	324		
什么是注册表.....	325		
如何在 Delphi 中对注册表进行操作	325		
如何将二进制数据添加到注册表.....	326		

第 1 章

IDE 应用

- ☞ 如何向项目中添加和删除单元
- ☞ 如何实现控件面板的自动滚动
- ☞ 如何控制代码的执行位置
- ☞ 如何快速编译项目文件和 PAS 文件
- ☞ Delphi 源文件中都包括哪些类型的文件
- ☞ 如何保存自己的编程环境
- ☞ 如何将做好的界面添加到模板中
- ☞ Delphi 中包含哪些常用的快捷键

1

如何向项目中添加和删除单元

难度系数 ★★

人气指数 80%

专家解答

1. 添加单元

(1) 单击【Project】|【Add to Project】菜单项，将会出现如图 1-1 所示的对话框，选择一个文件，然后单击【打开】按钮就可以了。



图 1-1 【Add to Project】对话框

(2) 单击【View】|【Units】菜单项，将会看到刚才添加的单元已经被程序引用了，如图 1-2 所示。

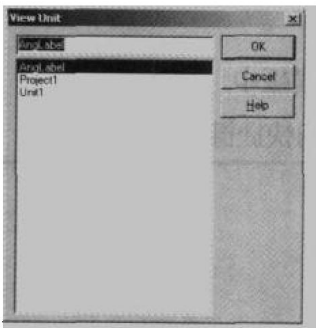


图 1-2 程序中包含了刚才添加的单元

2. 删除单元

单击【Project】|【Remove from Project】菜单项，将会出现如图 1-3 所示的对话框，其中显示了程序使用的所有单元，这里选择 AngLabel，然后单击【OK】按钮就可以了。

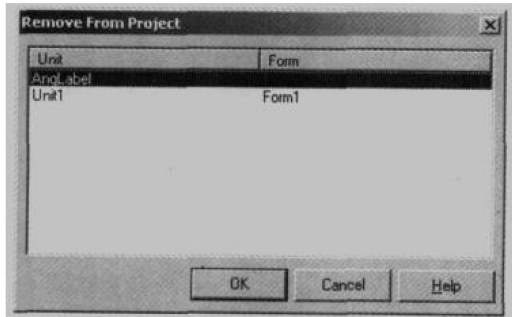


图 1-3 【Remove from Project】对话框

注意，这里并没有把单元文件从硬盘上物理删除，只是在项目文件中删除了该单元的信息，使其对

于这个项目不可见，就好像被删除了一样。要想从硬盘中删除该单元，需要从资源管理器中删除它。

2

如何实现控件面板的自动滚动

难度系数 ★★★

人气指数 85%

问题详述

在 Delphi 的控件面板上有许多的控件，有时候要查找到自己所需要的控件，要不停地单击鼠标，既繁琐又费时，除非自定义控件面板，隐藏掉多余的暂时不用的控件，当使用的时候再调出来，也非常麻烦。如果能实现控件面板的自动滚动，那就可以大大加快编程速度了。

专家解答

Borland 开发小组提供了一个小技巧，可以通过修改注册表来实现控件自动滚动的功能，下面说明实现方法。

- (1) 建立一个扩展名为 reg 的注册表项文件。
- (2) 用 Windows 的写字板来进行编辑。文件里的具体内容如下：

```
REGEDIT4
[HKEY_CURRENT_USER\Software\Borland\Delphi6.0\Extras]
"AutoPaletteSelect"="1"
"AutoPaletteScroll"="1"
```

- (3) 退出并保存后，双击运行这个注册表项文件。

(4) 再次启动重新 Delphi 7，当鼠标停留在面板的左右两端的时候，所有的控件标签页会自动向前或者向后滚动，省去单击鼠标的操作；当鼠标停留在某个控件标签页上的时候，会自动显示这一页面的所有控件。这两个选项使 Delphi 的控件面板活了起来，操作起来十分地方便。

3

如何控制代码的执行位置

难度系数 ★★

人气指数 95%

专家解答

- (1) 利用 Step Over 命令 (F8) 或 Trace Into 命令 (F7) 可以逐行执行代码。
 - (2) Trace Into 命令不会进入过程或函数的内部。一般是在遇到断点后需要恢复运行时才使用这个命令。
 - (3) Run to Cursor (F4) 命令可以让 Delphi 执行到光标所在的行。如果要绕过一个循环或许多代码，F4 键是很有用的，而 F7 键和 F8 键就不太合适。
- 记住，在代码编辑器中，任何时候都可以设置断点，即使当程序正在运行时也是如此。所以，不必事先把所有断点都设好。
- (4) 要临时使程序暂停，可以使用 Program Pause 命令。如果要判断程序是否陷于死循环中，这个命令比较有用。

4

快速编译项目和 PAS 文件

难度系数 ★★★

人气指数 80%

专家解答

1. 快速编译项目

在不打开 Delphi 程序的情况下，也可以对项目文件进行快速编译。这时需要修改注册表，通过注

册表把 Delphi 的项目文件和 Delphi 编译器直接关联起来，注册表项文件的内容如下：

REGEDIT4

```
[HKEY_CLASSES_ROOT\DelphiProject]
@="Delphi 7 Project"
"EditFlags"=hex:00,00,00,00
"AlwaysShowExt"=""
[HKEY_CLASSES_ROOT\DelphiProject\DefaultIcon]
@="F:\Program Files\Borland\Delphi 7\Bin\Delphi32.exe, 4"
[HKEY_CLASSES_ROOT\DelphiProject\Shell]
@=""
[HKEY_CLASSES_ROOT\DelphiProject\Shell\Open]
@="&Open"
"EditFlags"=hex:01,00,00,00
[HKEY_CLASSES_ROOT\DelphiProject\Shell\Open\command]
@="F:\Program Files\Borland\Delphi 7\Bin\delphi32.exe /ns %1"
[HKEY_CLASSES_ROOT\DelphiProject\Shell\Open\ddeexec]
@="[open(\"%1\")]"
[HKEY_CLASSES_ROOT\DelphiProject\Shell\Open\ddeexec\Application]
@="DELPHI32"

[HKEY_CLASSES_ROOT\DelphiProject\Shell\AutoBuild]
"EditFlags"=hex:01,00,00,00
[HKEY_CLASSES_ROOT\DelphiProject\Shell\AutoBuild\command]
@="F:\Program Files\Borland\Delphi 7\Bin\delphi32.exe /ns %1 /b"
[HKEY_CLASSES_ROOT\DelphiProject\Quick View]
@="**"
```

注意：Delphi 的目录根据用户安装的目录而定。编辑完毕，保存后，双击运行这个注册表项文件，再重新启动计算机。然后在资源管理器中，选择一个 Delphi 的项目文件，右键单击该项目文件，将会弹出编译菜单选项，可以直接生成 Exe 文件，如图 1-4 所示。

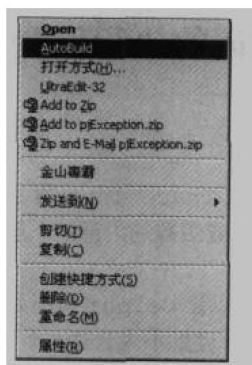


图 1-4 右键菜单添加的自动编译的选项

2. 快速编译 PAS 文件

在没有打开 Delphi 的情况下可以直接编译 Delphi 的 PAS 源代码，要实现这种功能需要编辑注册表，通过注册表把 Delphi 的 PAS 源代码文件和 Delphi 编译器直接关联起来，下面说明操作方法。

- (1) 新建一个文件，将其扩展名改为 .reg。
- (2) 使用记事本打开该文件，向其中添加文字如下：

REGEDIT4

```
[HKEY_CLASSES_ROOT\DelphiUnit]
@="Delphi Source File"
"EditFlags"=hex:00,00,00,00
"AlwaysShowExt"=""
```

```
[HKEY_CLASSES_ROOT\DelphiUnit\DefaultIcon]
@="F:\Program Files\Borland\Delphi 7\Bin\Delphi32.exe,5"

[HKEY_CLASSES_ROOT\DelphiUnit\Shell]
@=""

[HKEY_CLASSES_ROOT\DelphiUnit\Shell\Open]
@="&Open"

[HKEY_CLASSES_ROOT\DelphiUnit\Shell\Open\command]
@="F:\Program Files\Borland\Delphi 7\Bin\Delphi32.exe /ns /np"

[HKEY_CLASSES_ROOT\DelphiUnit\Shell\Open\ddeexec]
@="{open(\"%1\")}"

[HKEY_CLASSES_ROOT\DelphiUnit\Shell\Open\ddeexec\Application]
@="DELPHI32"

[HKEY_CLASSES_ROOT\DelphiUnit\Shell\Compile2DCU]
"EditFlags"=hex:01,00,00,00

[HKEY_CLASSES_ROOT\DelphiUnit\Shell\Compile2DCU\command]
@="F:\Program Files\Borland\Delphi 7\Bin\dcc32.exe %1"

[HKEY_CLASSES_ROOT\DelphiUnit\QuickView]
@="*"

```

(3) 编辑完毕, 保存后, 双击运行这个注册表项文件, 再重新启动计算机。然后在资源管理器中, 选择一个 Delphi 的 PAS 源代码文件, 右键单击该项目文件, 将会弹出编译菜单选项, 可以直接生成 DCU 文件, 如图 1-5 所示。

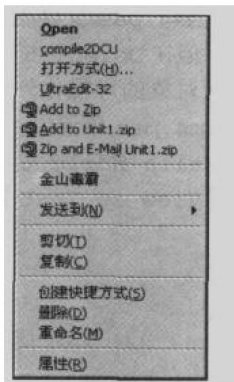


图 1-5 右键菜单中添加的快速编译功能

5 Delphi 源文件中包括哪些文件, 各自的作用如何

难度系数 ★★

人气指数 100%

专家解答

Delphi 是以工程为中心的开发产品。这意味着每个应用程序都是一个工程, 由一个或多个文件以及工程文件组成。组成工程的几种文件包括: 源代码、窗体、编译过的单元、配置、选项、包以及备份文件, 下面介绍各个文件的作用。

1. 工程文件 (.dpr)

工程文件具有 .dpr 扩展名, 在本质上它包含了应用程序的入口点, 位于 begin 和 end 之间。简单的 "HelloWorld" 程序的工程文件如下所示:

HelloWorld.dpr

```

program HelloWorld;

uses
  Forms,
  umain in 'umain.pas' {Form1};

{$R *.RES}

begin
  Application.Initialize;
  Application.CreateForm(TForm1, Form1);
  Application.Run;
end.

```

从【Project】菜单选择【View Source】可以看到工程源代码。上面列出的代码都是由 Delphi 自动添加的。program 语句指出了可执行文件的名字。uses 子句后部是逗号分隔的列表，包含了所有显式包括在工程中的文件。\$R 语句是编译器指令。编译器指令 | \$R *.RES | 指示 Delphi 在与工程同名、扩展名为 .RES 的文件中，查找 Windows 资源信息。begin 与 end. 对应 C 中的 main() 子程序、Visual Basic 的起始过程等价。典型的 Delphi 应用程序以 Application.Initialize 开始，以 Application.Run 结束。

注意：修改 .dpr 文件中的代码是可以的，但通常这并不必要。除非确实有修改它的原因，否则最好还是让 Delphi 来管理工程源文件。

2. 源代码文件 (.pas)

Pascal 文件具有 .pas 扩展名。通常每个工程至少有一个单元，单元供编写代码之用。如果创建了窗体或数据模块，将会得到 .dfm 和 .pas 两个文件。

3. 窗体与数据模块 (.dfm)

窗体与数据模块具有 .dfm 扩展名，并且与 .pas 文件相关联。代码实际写在具有 .pas 扩展名的源代码单元中。dfm 文件原来是二进制文件，但在 Delphi 5 之后已成为脚本化的文本文件，其中定义了一些资源，使得窗体和数据模块能够存储对象的可视化外观。窗体是 TForm 的子类，数据模块是 TDataModule 的子类，二者都定义在 forms.pas 单元中。

如果想看一看窗体的持久化脚本，在 Hello World 应用程序中先把窗体置于前台，在窗体上单击鼠标右键，显示窗体上下文菜单，并选择【View as Text】。main.dfm 的资源脚本列出如下：

```

main.dfm
object Form1: TForm1
  Left = 244
  Top = 138
  Width = 783
  Height = 540
  Caption = 'Form1'
  Color = clBtnFace
  Font.Charset = DEFAULT_CHARSET
  Font.Color = clWindowText
  Font.Height = -13
  Font.Name = 'MS Sans Serif'
  Font.Style = []
  OldCreateOrder = False
  PixelsPerInch = 120
  TextHeight = 16
object RadioGroup1: TRadioGroup
  Left = 296
  Top = 208
  Width = 201
  Height = 193
  Caption = 'Customary Greetings'
  ItemIndex = 0

```

```

Items.Strings = (
  'Geek'
  'PC Gamer'
  'English'
  'German')
TabOrder = 0
OnClick = RadioGroup1Click
end
end

```

说明：在上面列出的文本形式的 main.dfm 结尾处，可以看到在例子程序中输入的 Item.Strings 数据。这就是 .dfm 文件，它存储了一些描述窗体文件和数据模块的数据。

4. 配置与选项文件 (.dof)

当改动【Project Options】设置时，Delphi 把改动存储在一个具有 .dof 扩展名的文件中。当所做的改动影响到了应用程序的编译方式时，这些改动将以文本形式存储到 .cfg 文件中或配置文件中。

实际上分别生成了几个文件来存储配置、选项、To-Do 和其他类型的信息。如果在【Help Topics】对话框的【Find】属性页中查找【Generated Files】，即可得到关于存储工程信息的所有生成文件的信息。大部分情况下，Delphi 自动管理这些文件，请勿删除它们。

无需记住所有的文件及其作用，只需记住一条规则：只能删除扩展名中带有~（波浪线）的文件、.dcu 文件或者不需要的文件。如果使用了 SourceSafe 这样的版本控制产品，即使误删了需要的文件，也可从 SourceSafe 中将其恢复。

5. 中间编译单元 (.dcu)

编译过的单元是不可执行的，以 .dcu 为扩展名。在建立应用程序的链接阶段，所有的 .dcu 文件链接起来成为可执行程序。这些文件是可以删除的，但最好还是让 Delphi 来管理这些文件。

注意：如果只分发 DCU 文件，编译过的代码在 Delphi 将来发布的版本中可能是无效的。如果不升级，程序员就无法继续使用这些 DCU 文件。如果出售的是工具软件，可以考虑公开源代码并在许可协议中规定可接受的使用方式。

建立应用程序时，Delphi 会把源代码文件与编译过的单元进行比较。如果源代码没有被修改过，Delphi 就不需要重新编译源文件。如果希望其他的开发者使用这种产品来建立应用程序，可以只发布 .dcu 文件，而不发布源文件。以这种方式，其他的开发者可以使用该代码，而无需确切知道代码是如何编写的。发布 .dcu 文件确实是一种方法，这样就可以传播私有代码而无需将其公开。

6. 备份文件 (.~pas)

无论何时，只要改动了一个文件并且进行了保存，Delphi 都会对该文件已存在的版本进行备份。这样，就拥有该文件的最新版本以及一个稍早一点的版本。对备份文件的命名惯例是：它的名字与原文件几乎相同，但在‘.’与扩展名的第一个字母之间插入了一个~（波浪线）。例如，main.pas 就变成了 main.~pas。

由于备份文件不会以其他方式修改，因此要恢复备份，只要在资源管理器中重命名备份文件，去掉其名字中的波浪线即可。如果开始开发后一直使用某种版本控制产品，在更新文件的已归档版本时经常保存，那么不会丢掉任何修改。

7. 应用程序文件 (.dll, .exe, .ocx)

读者可能已经熟悉了几种基本类型的应用程序文件，Delphi 都能够创建所有种类的应用程序文件。包括动态链接库 (.dll)、可执行文件 (.exe)、ActiveX 控件 (.ocx，因为以前 ActiveX 支持的是 OLE 命名规范)。

其中每一种文件都代表一种终端产品，又经过编译和链接的代码组成。可执行应用程序是单独运行的程序或进程外服务器。动态链接库代表资源文件或进程内服务器，而 ActiveX 控件是用于建立其他程序的支持性控件。

6

如何保存自己的编程环境

难度系数 ★★

人气指数 95%

问题详述

每个人都有自己的爱好，习惯于不同的编程环境，能不能在 Delphi 中保存自己的编程环境，当程序再次启动时，直接设置好编程环境呢？

专家解答

(1) 启动 Delphi 后，窗口设置如图 1-6 所示。

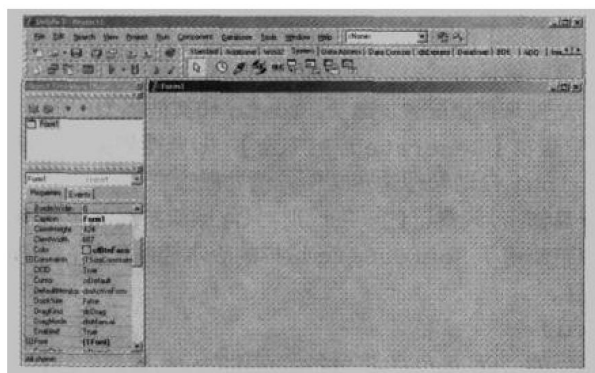


图 1-6 程序的窗口设置

(2) 单击【View】|【Desktop】|【Save Desktop】菜单项，将会出现如图 1-7 所示的窗口，在这里可以设定自定义桌面的名称。

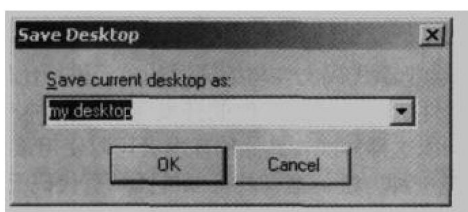


图 1-7 设定自定义桌面的名称

(3) 关闭 Delphi，再次启动它。单击【View】|【Desktop】菜单项，将会出现子菜单如图 1-8 所示。单击其中的【my desktop】菜单项，就会将 Delphi 窗口恢复到如图 1-6 所示的状态。

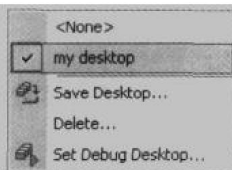


图 1-8 【Desktop】菜单下的子菜单

7

如何将做好的界面添加到模板中

难度系数 ★★★

人气指数 75%

问题详述

编程过程中，经常会重复使用某一个界面，比如使用类似的向导窗口、相同的对话框、信息提示框等。那么能不能把这些类似的界面保存为模板，以便日后调用呢？

专家解答

Delphi 提供了一个很好的机制,使用这种机制可以快速将窗体保存为模板,并可以在【New Items】对话框中直接调用。

(1) 新建一个项目,向窗体中添加两个 Button 控件,如图 1-9 所示。

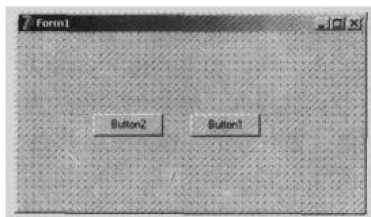


图 1-9 程序使用的控件

(2) 在窗体中右击鼠标,将会弹出快捷菜单,如图 1-10 所示。

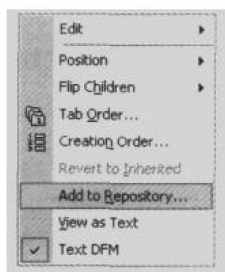


图 1-10 弹出的快捷菜单

(3) 单击图 1-10 中的【Add to Repository】菜单项,将会出现如图 1-11 所示的【Add to Repository】对话框,这里将各个文本框设置如图 1-11 所示。设置完毕后,单击【OK】按钮。

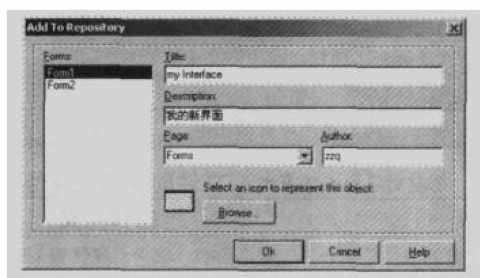


图 1-11 【Add to Repository】对话框

(4) 单击【File】|【New】|【Others】菜单项,将会出现【New Items】对话框,这里选择【Forms】标签页,可以看到刚才创建的模板出现在窗口中,如图 1-12 所示。

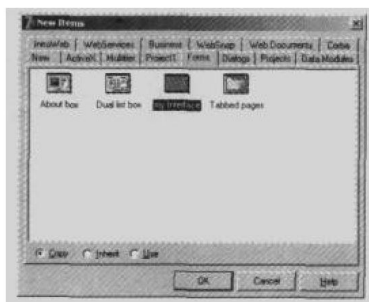


图 1-12 【New Items】对话框