

新版软件参考手册系列

# PowerBuilder

## 6.0~6.5参考手册

(语言、函数、对象篇)

新世纪工作室 编著



6-62  
1/2

国防工业出版社

TP311.56-62

573

新版软件参考手册系列

# PowerBuilder 6.0~6.5 参考手册

(语言、函数、对象篇)

新世纪工作室 编著



国防工业出版社

·北京·

003916

图书在版编目(CIP)数据

PowerBuilder6.0~6.5 参考手册:语言、函数、对象  
篇/新世纪工作室编著. - 北京:国防工业出版社,1999.8  
(新版软件参考手册系列)

ISBN 7-118-02087-7

I. P… II. 新… III. 数据库管理系统-软件工具, PowerBuilder6.0~6.5 IV. TP311.56

中国版本图书馆 CIP 数据核字(1999)第 18535 号

JS356/01

国防工业出版社出版发行

(北京市海淀区紫竹院南路 23 号)

(邮政编码 100044)

北京怀柔新华印刷厂印刷

新华书店经售

\*

开本 787×1092 1/16 印张 25¼ 578 千字

1999 年 8 月第 1 版 1999 年 8 月北京第 1 次印刷

印数:1-4000 册 定价:34.00 元

---

(本书如有印装错误,我社负责调换)

## 序

计算机软件的应用基本上可以分成两个层次。对于纯操作型软件,如 Word、Frong-Page、WPS 等,通常针对非专业计算机用户。有了这些软件,只需经过有限步骤的操作,就可以直接满足某些应用需求。而另一类软件,如 Visual Basic、Visual C++、Power-Builder 等,则要复杂得多。这些软件通常是供专业计算机用户(程序员)使用的,不仅要熟悉软件的环境与操作,还要掌握软件为应用开发所提供的编程元素,如特定的编程语言、内置的资源 and 工具等,这样才可能在其基础上开发出可以实现最终目标的应用系统。所以,对于后一类软件,专业用户或希望成为专业用户的初学者往往要投入大量时间和精力来查阅和尝试软件语言、资源以及工具的用法。

另外,软件更新换代的速度很快,新版软件总是会修订旧版软件的功能,并增添新的功能。为此,专业计算机用户还需要跟踪最新软件技术,并在很短时间内掌握最新软件工具的使用。

因此,根据这种需要,这类软件的相关书籍必须具备下列几个特点:

(1)基于最新版本。新版软件往往兼容相应的旧版软件,因此新版软件的相关书籍也基本适用于旧版本。

(2)内容全面。特定的应用开发往往涉及软件提供的各种资源,因此要求专业计算机用户对软件的掌握必须全面、深入。

(3)结构合理,便于查阅。随着软件规模不断扩大,其复杂性也不断提高,合理的组织与编排可以为专业计算机用户节省大量时间。

目前市面上虽然出版了不少技术参考读物,但其内容基本上来源于相应的英文书籍或者软件的帮助文本。一方面,译者的英语水平和软件的掌握程度对书籍的质量影响极大;另一方面,对零碎的帮助文本组织得是否合理也是面临的一大问题。如果再考虑到软件的更新,以致质量很难有保证。因此,目前这类书籍的现状是质量参差不齐,上乘者极少。

针对这种情况,本着为软件开发和应用人员服务的原则,我们组织了一批经验丰富的软件工程师,编写了《新版软件参考手册系列》丛书。

这套丛书针对目前最新版本的流行软件,以软件开发环境、资源和工具为基础,主要以专业软件开发人员为读者对象,所涉及的软件基本包括 PowerBuilder 6.5、SQL Server 7.0、Novell Netware 5.0、Visual Basic 6.0 和 Visual C++ 6.0。这套丛书列举并举例说明了这些软件开发工具的组成元素,各条目间相关性小,但自成一体。全书强调全面、详尽、便于查阅,因为它是供专业软件开发人员在实际开发过程中参考查阅用的,非专业人员可能不需要这套丛书,但对于专业开发者,它却是必不可少的。

我们相信,丛书的出版必将为广大读者开辟跟踪、掌握、运用、进而创造新技术的最佳途径。

当然,尽管我们强调以质取胜,宁缺勿滥,但是,由于相应软件实在极其复杂,加上时间较紧,因此书中可能还难免有一些错误,请各位读者不吝指正。

# 前 言

PowerBuilder 是 Sybase 公司的子公司 PowerSoft 推出的一个用于设计企业级客户/服务器应用程序的开发工具,它支持多种平台,具备跨平台开发应用程序的能力。本书是 PowerBuilder 的语言、函数、对象参考手册,详细介绍了 PowerScript 语言、通用系统函数以及对象的属性、事件和函数。

全书共分 4 章。第一章完整地介绍了 PowerScript 语言,内容包括数据类型、变量声明、赋值语句、条件语句、循环语句、嵌入式 SQL 语句等,详细说明了应用程序中动态创建和删除对象的方法。第二章介绍了通用的系统函数;第三章完整地介绍了 PowerBuilder 中各种对象的属性、事件和函数;第四章详细介绍了数据窗口对象的属性、应用程序访问这些属性的方法以及数据窗口画笔函数。另外,对于功能、语法、用法完全相同的对象函数,在附录中单独介绍,而对存在某些细微差别的函数,依然作为单独条目在各对象中介绍。

本书还给出了可直接在应用程序中运行的大量示例,供用户编写应用程序时参考。

本书适合 PowerBuilder 开发人员阅读,是开发 PowerBuilder 应用程序不可缺少的参考书。如果与本书的姊妹篇《PowerBuilder 6.0 ~ 6.5 参考手册(控件篇)》搭配使用,则效果会更好。

# 目 录

|                                                  |    |                                |    |
|--------------------------------------------------|----|--------------------------------|----|
| 第一章 PowerScript 语言 .....                         | 1  | 1.5.2 释放对象 .....               | 23 |
| 1.1 语言基础 .....                                   | 1  | 1.5.3 收集废弃对象 .....             | 24 |
| 1.1.1 续行 .....                                   | 1  | 1.6 赋值语句 .....                 | 24 |
| 1.1.2 注释 .....                                   | 2  | 1.7 条件语句 .....                 | 25 |
| 1.1.3 标识符 .....                                  | 2  | 1.7.1 IF 语句 .....              | 25 |
| 1.1.4 保留字 .....                                  | 3  | 1.7.2 CHOOSE CASE 语句 .....     | 26 |
| 1.1.5 特殊字符 .....                                 | 3  | 1.8 循环语句 .....                 | 27 |
| 1.1.6 空值 .....                                   | 4  | 1.8.1 DO ... LOOP 循环 .....     | 27 |
| 1.1.7 对象、属性、函数和事件<br>的引用 .....                   | 5  | 1.8.2 FOR ... NEXT 循环 .....    | 28 |
| 1.1.8 This、Parent、Parent Window<br>和 Super ..... | 5  | 1.8.3 GOTO 语句 .....            | 29 |
| 1.2 数据类型 .....                                   | 6  | 1.8.4 EXIT(退出循环) .....         | 30 |
| 1.2.1 标准数据类型 .....                               | 6  | 1.8.5 CONTINUE(继续<br>循环) ..... | 30 |
| 1.2.2 Any 类型 .....                               | 10 | 1.8.6 循环嵌套 .....               | 30 |
| 1.2.3 系统对象数据类型 .....                             | 10 | 1.9 返回语句与终止程序运行 .....          | 31 |
| 1.2.4 枚举类型 .....                                 | 11 | 1.9.1 RETURN 语句 .....          | 31 |
| 1.3 声明语句及变量作用域 .....                             | 12 | 1.9.2 HALT 语句 .....            | 31 |
| 1.3.1 声明变量 .....                                 | 12 | 1.10 CALL 语句 .....             | 31 |
| 1.3.2 声明数组 .....                                 | 13 | 1.11 嵌入式 SQL 语句 .....          | 32 |
| 1.3.3 声明常量 .....                                 | 14 | 1.11.1 建立和断开与数据库<br>的连接 .....  | 33 |
| 1.3.4 声明外部函数 .....                               | 14 | 1.11.2 提交与回滚事务 .....           | 33 |
| 1.3.5 声明数据库存储过程 .....                            | 16 | 1.11.3 单行检索语句<br>SELECT .....  | 34 |
| 1.3.6 变量的作用域 .....                               | 17 | 1.11.4 插入语句 INSERT .....       | 35 |
| 1.4 操作符 .....                                    | 19 | 1.11.5 删除语句 DELETE .....       | 35 |
| 1.4.1 算术操作符 .....                                | 19 | 1.11.6 修改语句 UPDATE .....       | 35 |
| 1.4.2 关系操作符 .....                                | 20 | 1.11.7 检查 SQL 语句执行<br>情况 ..... | 35 |
| 1.4.3 逻辑操作符 .....                                | 21 | 1.11.8 读取多行数据 .....            | 36 |
| 1.4.4 连接操作符 .....                                | 21 | 1.11.9 动态 SQL .....            | 38 |
| 1.4.5 操作符的优先级 .....                              | 21 | 1.11.10 SELECTBLOB 语句 .....    | 42 |
| 1.5 创建与释放对象实例 .....                              | 22 |                                |    |
| 1.5.1 创建对象 .....                                 | 23 |                                |    |

|                              |    |                                |    |
|------------------------------|----|--------------------------------|----|
| 1.12 应用程序开发流程简介 .....        | 42 | 2.4.12 Today( ) .....          | 66 |
| <b>第二章 系统函数</b> .....        | 45 | 2.4.13 Year( ) .....           | 67 |
| 2.1 数组函数 .....               | 45 | 2.5 DDE 客户函数 .....             | 67 |
| 2.1.1 LowerBound( ) .....    | 46 | 2.5.1 CloseChannel( ) .....    | 68 |
| 2.1.2 UpperBound( ) .....    | 46 | 2.5.2 ExecRemote( ) .....      | 69 |
| 2.2 Blob(大二进制对象)函数 .....     | 47 | 2.5.3 GetDataDDE( ) .....      | 70 |
| 2.2.1 Blob( ) .....          | 47 | 2.5.4 GetDataDDEOrigin( ) ...  | 70 |
| 2.2.2 BlobEdit( ) .....      | 48 | 2.5.5 GetRemote( ) .....       | 70 |
| 2.2.3 BlobMid( ) .....       | 48 | 2.5.6 OpenChannel( ) .....     | 71 |
| 2.2.4 Len( ) .....           | 49 | 2.5.7 RespondRemote( ) .....   | 72 |
| 2.2.5 String( ) .....        | 49 | 2.5.8 SetRemote( ) .....       | 72 |
| 2.3 数据类型检查与转换函数 .....        | 50 | 2.5.9 StartHotLink( ) .....    | 74 |
| 2.3.1 Asc( ) .....           | 50 | 2.5.10 StopHotLink( ) .....    | 74 |
| 2.3.2 Char( ) .....          | 51 | 2.6 DDE 服务器函数 .....            | 75 |
| 2.3.3 Dec( ) .....           | 51 | 2.6.1 GetCommandDDE( ) .....   | 75 |
| 2.3.4 Double( ) .....        | 52 | 2.6.2 GetCommandDDEOrigin      |    |
| 2.3.5 Integer( ) .....       | 52 | ( ) .....                      | 76 |
| 2.3.6 Long( ) .....          | 52 | 2.6.3 GetDataDDE( ) .....      | 76 |
| 2.3.7 Real( ) .....          | 53 | 2.6.4 GetDataDDEOrigin( ) ...  | 76 |
| 2.3.8 Date( ) .....          | 54 | 2.6.5 RespondRemote( ) .....   | 77 |
| 2.3.9 DateTime( ) .....      | 55 | 2.6.6 SetDataDDE( ) .....      | 77 |
| 2.3.10 IsDate( ) .....       | 56 | 2.6.7 StartServerDDE( ) .....  | 77 |
| 2.3.11 IsNull( ) .....       | 56 | 2.6.8 StopServerDDE( ) .....   | 78 |
| 2.3.12 IsNumber( ) .....     | 56 | 2.7 文件操作函数 .....               | 78 |
| 2.3.13 IsTime( ) .....       | 57 | 2.7.1 FileClose( ) .....       | 79 |
| 2.3.14 String( ) .....       | 57 | 2.7.2 FileDelete( ) .....      | 79 |
| 2.3.15 Time( ) .....         | 60 | 2.7.3 FileExists( ) .....      | 80 |
| 2.4 日期、时间函数 .....            | 61 | 2.7.4 FileLength( ) .....      | 80 |
| 2.4.1 Day( ) .....           | 62 | 2.7.5 FileOpen( ) .....        | 81 |
| 2.4.2 DayName( ) .....       | 62 | 2.7.6 FileRead( ) .....        | 82 |
| 2.4.3 DayNumber( ) .....     | 63 | 2.7.7 FileSeek( ) .....        | 83 |
| 2.4.4 DaysAfter( ) .....     | 63 | 2.7.8 FileWrite( ) .....       | 84 |
| 2.4.5 Hour( ) .....          | 63 | 2.7.9 GetFileOpenName( ) ..... | 85 |
| 2.4.6 Minute( ) .....        | 64 | 2.7.10 GetFileSaveName( ) ...  | 86 |
| 2.4.7 Month( ) .....         | 64 | 2.8 国际化函数 .....                | 87 |
| 2.4.8 Now( ) .....           | 65 | 2.8.1 IsAllArabic( ) .....     | 87 |
| 2.4.9 RelativeDate( ) .....  | 65 | 2.8.2 IsAllHebrew( ) .....     | 87 |
| 2.4.10 RelativeTime( ) ..... | 65 | 2.8.3 IsAnyArabic( ) .....     | 88 |
| 2.4.11 Second( ) .....       | 66 | 2.8.4 IsAnyHebrew( ) .....     | 88 |

|         |                           |     |         |                    |     |
|---------|---------------------------|-----|---------|--------------------|-----|
| 2.8.5   | IsArabic( )               | 88  | 2.11.3  | PrintCancel( )     | 107 |
| 2.8.6   | IsArabicAndNumbers<br>( ) | 89  | 2.11.4  | PrintClose( )      | 107 |
| 2.8.7   | IsHebrew( )               | 89  | 2.11.5  | PrintDataWindow( ) | 108 |
| 2.8.8   | IsHebrewAndNumbers<br>( ) | 89  | 2.11.6  | PrintDefineFont( ) | 108 |
| 2.8.9   | Reverse( )                | 90  | 2.11.7  | PrintLine( )       | 109 |
| 2.8.10  | ToAnsi( )                 | 90  | 2.11.8  | PrintOpen( )       | 110 |
| 2.8.11  | ToUnicode( )              | 90  | 2.11.9  | PrintOval( )       | 111 |
| 2.9     | 库管理函数                     | 91  | 2.11.10 | PrintPage( )       | 111 |
| 2.9.1   | LibraryCreate( )          | 91  | 2.11.11 | PrintRect( )       | 112 |
| 2.9.2   | LibraryDelete( )          | 92  | 2.11.12 | PrintRoundRect( )  | 113 |
| 2.9.3   | LibraryDirectory( )       | 92  | 2.11.13 | PrintScreen( )     | 113 |
| 2.9.4   | LibraryExport( )          | 93  | 2.11.14 | PrintSend( )       | 114 |
| 2.9.5   | LibraryImport( )          | 94  | 2.11.15 | PrintSetFont( )    | 115 |
| 2.10    | 数值计算函数                    | 95  | 2.11.16 | PrintSetSpacing( ) | 116 |
| 2.10.1  | Abs( )                    | 96  | 2.11.17 | PrintSetup( )      | 116 |
| 2.10.2  | Ceiling( )                | 96  | 2.11.18 | PrintText( )       | 117 |
| 2.10.3  | Cos( )                    | 97  | 2.11.19 | PrintWidth( )      | 118 |
| 2.10.4  | Exp( )                    | 97  | 2.11.20 | PrintX( )          | 118 |
| 2.10.5  | Fact( )                   | 97  | 2.11.21 | PrintY( )          | 119 |
| 2.10.6  | Int( )                    | 98  | 2.12    | 注册函数               | 119 |
| 2.10.7  | Log( )                    | 98  | 2.12.1  | RegistryDelete( )  | 120 |
| 2.10.8  | LogTen( )                 | 98  | 2.12.2  | RegistryGet( )     | 120 |
| 2.10.9  | Max( )                    | 99  | 2.12.3  | RegistryKeys( )    | 121 |
| 2.10.10 | Min( )                    | 99  | 2.12.4  | RegistrySet( )     | 122 |
| 2.10.11 | Mod( )                    | 100 | 2.12.5  | RegistryValues( )  | 122 |
| 2.10.12 | Pi( )                     | 100 | 2.13    | 字符串操作函数            | 123 |
| 2.10.13 | Rand( )                   | 101 | 2.13.1  | Fill( )            | 124 |
| 2.10.14 | Randomize( )              | 101 | 2.13.2  | Left( )            | 124 |
| 2.10.15 | Round( )                  | 101 | 2.13.3  | LeftTrim( )        | 125 |
| 2.10.16 | Sign( )                   | 102 | 2.13.4  | Len( )             | 125 |
| 2.10.17 | Sin( )                    | 102 | 2.13.5  | Lower( )           | 125 |
| 2.10.18 | Sqrt( )                   | 102 | 2.13.6  | Match( )           | 126 |
| 2.10.19 | Tan( )                    | 103 | 2.13.7  | Mid( )             | 127 |
| 2.10.20 | Truncate( )               | 103 | 2.13.8  | Pos( )             | 128 |
| 2.11    | 打印函数                      | 104 | 2.13.9  | Replace( )         | 128 |
| 2.11.1  | Print( )                  | 105 | 2.13.10 | Right( )           | 129 |
| 2.11.2  | PrintBitmap( )            | 106 | 2.13.11 | RightTrim( )       | 129 |
|         |                           |     | 2.13.12 | Space( )           | 130 |
|         |                           |     | 2.13.13 | Trim( )            | 130 |

|         |                      |     |                        |                  |     |
|---------|----------------------|-----|------------------------|------------------|-----|
| 2.13.14 | Upper( )             | 130 | 2.17.9                 | PixelsToUnits( ) | 159 |
| 2.14    | 系统与环境函数              | 131 | 2.17.10                | PopulateError( ) | 159 |
| 2.14.1  | Clipboard( )         | 131 | 2.17.11                | RGB( )           | 160 |
| 2.14.2  | CommandParm( )       | 132 | 2.17.12                | SetNull( )       | 161 |
| 2.14.3  | DoScript( )          | 133 | 2.17.13                | SetPointer( )    | 162 |
| 2.14.4  | GetApplication( )    | 133 | 2.17.14                | SignalError( )   | 162 |
| 2.14.5  | GetEnvironment( )    | 133 | 2.17.15                | UnitsToPixels( ) | 163 |
| 2.14.6  | GetFocus( )          | 134 | <b>第三章 对象的属性、事件和函数</b> | 165              |     |
| 2.14.7  | Post( )              | 135 | 3.1                    | 应用对象             | 165 |
| 2.14.8  | ProfileInt( )        | 136 | 3.1.1                  | 应用对象的属性          | 165 |
| 2.14.9  | ProfileString( )     | 137 | 3.1.2                  | 应用对象的事件          | 166 |
| 2.14.10 | Restart( )           | 137 | 3.1.3                  | 应用对象的函数          | 167 |
| 2.14.11 | Run( )               | 138 | 3.2                    | 数组边界对象           | 169 |
| 2.14.12 | Send( )              | 138 | 3.2.1                  | 数组边界对象的属性        | 169 |
| 2.14.13 | SetProfileString( )  | 139 | 3.2.2                  | 数组边界对象的函数        | 169 |
| 2.14.14 | ShowHelp( )          | 140 | 3.3                    | 类定义对象            | 170 |
| 2.14.15 | SignalError( )       | 141 | 3.3.1                  | 类定义对象的属性         | 170 |
| 2.14.16 | Yield( )             | 141 | 3.3.2                  | 类定义对象的函数         | 171 |
| 2.15    | 定时操作函数               | 142 | 3.4                    | 上下文信息对象          | 173 |
| 2.15.1  | CPU( )               | 142 | 3.4.1                  | 上下文信息对象的属性       | 173 |
| 2.15.2  | Idle( )              | 143 | 3.4.2                  | 上下文信息对象的事件       | 173 |
| 2.15.3  | Timer( )             | 144 | 3.4.3                  | 上下文信息对象的函数       | 173 |
| 2.16    | 打开与关闭窗口              | 144 | 3.5                    | 上下文关键字对象         | 178 |
| 2.16.1  | Close( )             | 145 | 3.5.1                  | 上下文关键字对象的属性      | 179 |
| 2.16.2  | CloseWithReturn( )   | 146 | 3.5.2                  | 上下文关键字对象的事件      | 179 |
| 2.16.3  | Open( )              | 147 | 3.5.3                  | 上下文关键字对象的函数      | 179 |
| 2.16.4  | OpenSheet( )         | 148 | 3.6                    | C++对象            | 180 |
| 2.16.5  | OpenSheetWithParm( ) | 149 | 3.6.1                  | C++对象的属性         | 181 |
| 2.16.6  | OpenWithParm( )      | 150 | 3.6.2                  | C++对象的事件         | 181 |
| 2.17    | 其他系统函数               | 151 | 3.6.3                  | C++对象的函数         | 181 |
| 2.17.1  | Beep( )              | 152 | 3.7                    | 数据存储对象           | 181 |
| 2.17.2  | ClassName( )         | 153 | 3.7.1                  | 数据存储对象的属性        | 181 |
| 2.17.3  | DebugBreak( )        | 153 | 3.7.2                  | 数据存储对象的事件        | 182 |
| 2.17.4  | IntHigh( )           | 154 |                        |                  |     |
| 2.17.5  | IntLow( )            | 154 |                        |                  |     |
| 2.17.6  | IsValid( )           | 154 |                        |                  |     |
| 2.17.7  | KeyDown( )           | 155 |                        |                  |     |
| 2.17.8  | MessageBox( )        | 157 |                        |                  |     |

|                                    |     |                                     |     |
|------------------------------------|-----|-------------------------------------|-----|
| 3.7.3 数据存储对象的函数                    | 183 | 3.17 Inet 对象                        | 205 |
| 3.8 子数据窗口对象                        | 187 | 3.17.1 Inet 对象的属性                   | 205 |
| 3.9 动态描述区对象                        | 190 | 3.17.2 Inet 对象的事件                   | 206 |
| 3.9.1 动态描述区对象的<br>属性               | 190 | 3.17.3 Inet 对象的函数                   | 206 |
| 3.9.2 动态描述区对象的<br>事件               | 191 | 3.18 InternetResult 对象              | 208 |
| 3.9.3 动态描述区对象的<br>函数               | 191 | 3.18.1 InternetResult 对象<br>的属性     | 209 |
| 3.10 DynamicStagingArea 对象         | 197 | 3.18.2 InternetResult 对象<br>的事件     | 209 |
| 3.10.1 DynamicStagingArea<br>对象的事件 | 197 | 3.18.3 InternetResult 对象<br>的函数     | 209 |
| 3.10.2 DynamicStagingArea<br>对象的函数 | 197 | 3.19 ListViewItem 对象                | 210 |
| 3.11 枚举定义对象                        | 198 | 3.19.1 ListViewItem 对象<br>的属性       | 210 |
| 3.11.1 枚举定义对象的<br>属性               | 198 | 3.19.2 ListViewItem 对象<br>的函数       | 211 |
| 3.11.2 枚举定义对象的<br>函数               | 198 | 3.20 mailFileDescription 对象         | 211 |
| 3.12 枚举项定义对象                       | 199 | 3.20.1 mailFileDescription<br>对象的属性 | 211 |
| 3.12.1 枚举项定义对象的<br>属性              | 199 | 3.20.2 mailFileDescription<br>对象的函数 | 211 |
| 3.12.2 枚举项定义对象的<br>函数              | 199 | 3.21 邮件消息对象                         | 212 |
| 3.13 环境对象                          | 199 | 3.21.1 邮件消息对象的<br>属性                | 212 |
| 3.13.1 环境对象的属性                     | 199 | 3.21.2 邮件消息对象的<br>函数                | 212 |
| 3.13.2 环境对象的函数                     | 200 | 3.22 邮件接收者对象                        | 212 |
| 3.14 出错对象                          | 200 | 3.22.1 邮件接收者对象<br>的属性               | 213 |
| 3.14.1 出错对象的属性                     | 201 | 3.22.2 邮件接收者对象<br>的函数               | 213 |
| 3.14.2 出错对象的事件                     | 201 | 3.23 邮件会话对象                         | 213 |
| 3.14.3 出错对象的函数                     | 201 | 3.23.1 邮件会话对象的<br>属性                | 213 |
| 3.15 grAxis 对象                     | 202 | 3.23.2 邮件会话对象的<br>事件                | 213 |
| 3.15.1 grAxis 对象的属性                | 202 | 3.23.3 邮件会话对象的<br>函数                | 214 |
| 3.15.2 grAxis 对象的函数                | 203 | 3.24 MDIClient 对象                   | 226 |
| 3.16 grDispAttr 对象                 | 204 |                                     |     |
| 3.16.1 grDispAttr 对象<br>的属性        | 204 |                                     |     |
| 3.16.2 grDispAttr 对象<br>的函数        | 205 |                                     |     |

|                                   |     |                                        |     |
|-----------------------------------|-----|----------------------------------------|-----|
| 3.24.1 MDIClient 对象<br>的属性.....   | 226 | 3.31.1 管道对象的属性.....                    | 256 |
| 3.24.2 MDIClient 对象<br>的函数.....   | 226 | 3.31.2 管道对象的事件.....                    | 257 |
| 3.25 菜单对象.....                    | 227 | 3.31.3 管道对象的函数.....                    | 257 |
| 3.25.1 菜单对象的属性 .....              | 227 | 3.32 ProfileCall 对象 .....              | 260 |
| 3.25.2 菜单对象的事件.....               | 228 | 3.32.1 ProfileCall 对象的<br>属性.....      | 260 |
| 3.25.3 菜单对象的函数.....               | 228 | 3.32.2 ProfileCall 对象的<br>函数.....      | 260 |
| 3.26 MenuCascade 对象 .....         | 232 | 3.33 ProfileClass 对象.....              | 261 |
| 3.26.1 MenuCascade 对象<br>的属性..... | 233 | 3.33.1 ProfileClass 对象<br>的属性.....     | 261 |
| 3.26.2 MenuCascade 对象<br>的事件..... | 234 | 3.33.2 ProfileClass 对象<br>的函数.....     | 261 |
| 3.26.3 MenuCascade 对象<br>的函数..... | 234 | 3.34 ProfileLine 对象 .....              | 262 |
| 3.27 消息对象.....                    | 235 | 3.34.1 ProfileLine 对象<br>的属性.....      | 262 |
| 3.27.1 消息对象的属性.....               | 235 | 3.34.2 ProfileLine 对象<br>的函数.....      | 263 |
| 3.27.2 消息对象的事件.....               | 236 | 3.35 ProfileRoutine 对象 .....           | 264 |
| 3.27.3 消息对象的函数.....               | 236 | 3.35.1 ProfileRoutine 对象<br>的属性.....   | 264 |
| 3.28 OLEObject 对象 .....           | 236 | 3.35.2 ProfileRoutine 对象<br>的函数.....   | 265 |
| 3.28.1 OLEObject 对象<br>的事件.....   | 236 | 3.36 Profiling 对象 .....                | 267 |
| 3.28.2 OLEObject 对象<br>的函数.....   | 237 | 3.36.1 Profiling 对象的属性 ..              | 267 |
| 3.29 OLEStorage 对象 .....          | 243 | 3.36.2 Profiling 对象的函数 ..              | 268 |
| 3.29.1 OLEStorage 对象<br>的属性.....  | 243 | 3.37 ScriptDefinition 对象 .....         | 272 |
| 3.29.2 OLEStorage 对象<br>的事件.....  | 244 | 3.37.1 ScriptDefinition 对象<br>的属性..... | 272 |
| 3.29.3 OLEStorage 对象<br>的函数.....  | 244 | 3.37.2 ScriptDefinition 对象<br>的函数..... | 273 |
| 3.30 OLEStream 对象 .....           | 251 | 3.38 定时对象.....                         | 273 |
| 3.30.1 OLEStream 对象<br>的属性.....   | 251 | 3.38.1 定时对象的属性.....                    | 273 |
| 3.30.2 OLEStream 对象<br>的事件.....   | 251 | 3.38.2 定时对象的事件.....                    | 274 |
| 3.30.3 OLEStream 对象<br>的函数.....   | 252 | 3.38.3 定时对象的函数.....                    | 274 |
| 3.31 管道对象.....                    | 256 | 3.39 事务对象.....                         | 276 |
|                                   |     | 3.39.1 事务对象的属性.....                    | 276 |
|                                   |     | 3.39.2 事务对象的事件.....                    | 277 |
|                                   |     | 3.39.3 事务对象的函数.....                    | 277 |

|        |                                              |     |        |                            |     |
|--------|----------------------------------------------|-----|--------|----------------------------|-----|
| 3.40   | TreeViewItem 对象 .....                        | 279 | 4.1.5  | DataWindow 对象<br>的属性 ..... | 314 |
| 3.40.1 | TreeViewItem 对象<br>的属性 .....                 | 280 | 4.1.6  | 组框对象的属性 .....              | 319 |
| 3.40.2 | TreeViewItem 对象<br>的函数 .....                 | 280 | 4.1.7  | 直线对象的属性 .....              | 320 |
| 3.41   | 类型定义对象 .....                                 | 281 | 4.1.8  | 椭圆、矩形、圆角矩形<br>的属性 .....    | 321 |
| 3.41.1 | 类型定义对象的<br>属性 .....                          | 281 | 4.1.9  | 报表对象的属性 .....              | 322 |
| 3.41.2 | 类型定义对象的<br>函数 .....                          | 282 | 4.1.10 | TableBlob 对象的<br>属性 .....  | 323 |
| 3.42   | VariableCardinalityDe-<br>finition 对象 .....  | 282 | 4.1.11 | 文本对象的属性 .....              | 324 |
| 3.42.1 | VariableCardinalityDe-<br>finition 对象的属性 ... | 282 | 4.2    | 数据窗口对象属性的访问<br>方法 .....    | 326 |
| 3.42.2 | VariableCardinalityDe-<br>finition 对象的函数 ... | 282 | 4.2.1  | 数据窗口属性表达式 ...              | 326 |
| 3.43   | 变量定义对象 .....                                 | 283 | 4.2.2  | 数据窗口画笔表达式 ...              | 329 |
| 3.43.1 | 变量定义对象的<br>属性 .....                          | 283 | 4.2.3  | 数据窗口对象属性取值<br>及其类型 .....   | 330 |
| 3.43.2 | 变量定义对象的<br>函数 .....                          | 284 | 4.2.4  | 数据窗口对象属性表达<br>式的求值 .....   | 330 |
| 3.44   | 窗口对象 .....                                   | 285 | 4.2.5  | 数据窗口对象属性表达<br>式的出错处理 ..... | 330 |
| 3.44.1 | 窗口对象的属性 .....                                | 285 | 4.3    | 数据窗口画笔函数 .....             | 331 |
| 3.44.2 | 窗口对象的事件 .....                                | 287 | 4.3.1  | 聚合函数 .....                 | 331 |
| 3.44.3 | 窗口对象的函数 .....                                | 288 | 4.3.2  | 数据类型检查与转换<br>函数 .....      | 348 |
| 第四章    | 数据窗口对象 .....                                 | 304 | 4.3.3  | 数据窗口信息函数 .....             | 353 |
| 4.1    | 数据窗口对象的属性 .....                              | 304 | 4.3.4  | 日期、时间函数 .....              | 359 |
| 4.1.1  | 位图对象的属性 .....                                | 304 | 4.3.5  | 数值运算函数 .....               | 366 |
| 4.1.2  | 按钮对象的属性 .....                                | 305 | 4.3.6  | 字符串操作函数 .....              | 366 |
| 4.1.3  | 列对象的属性 .....                                 | 307 | 4.3.7  | 交叉表函数 .....                | 368 |
| 4.1.4  | 计算域的属性 .....                                 | 313 | 4.3.8  | 其他画笔函数 .....               | 370 |
|        |                                              |     | 附录     | 对象的公共函数 .....              | 374 |

# 第一章 PowerScript 语言

PowerScript 是 PowerBuilder 的编程语言,这是一种结构化的编程语言,用来编写函数和事件处理程序。PowerScript 提供了一套完整的嵌入式 SQL 语句,开发人员能够像使用其他语言成分(如赋值语句)那样使用它们,而且需要的话也可以使用特定数据库管理系统专用的 SQL 语句。需要嵌入式 SQL 语句时,无需逐字键入,而可利用 PowerBuilder 提供的可视化工具生成嵌入式 SQL 语句,然后把它粘贴到程序中。

PowerScript 不仅支持常规的数据类型,同时也支持用于处理多媒体数据的二进制大对象,当然也完全支持对象类型。本章将介绍 PowerScript 语言的概念、语法以及用法。

## 1.1 语言基础

每种编程语言都有自己的一组基本的语法约定,PowerBuilder 也不例外。PowerScript 是 PowerBuilder 的编程语言,我们编写的事件处理程序、用户自定义函数等都是用 PowerScript 语言编写的。这是一种自由格式的语言,语句行中的空格、缩进等格式编排信息完全被编译器忽略,为了阅读方便,可以随意安排语句行的位置。与其他大多数编程语言相似,PowerScript 提供了断行、续行的方法,如果愿意,在一行中写上几条语句也毫无问题。另外,该语言提供了两种注释方法,通过注释进一步提高程序的可读性。本节将简单介绍 PowerScript 语言的断行、续行、注释、标识符、代词、保留字等成分的格式和用法。

### 1.1.1 续行

通常情况下,PowerScript 的一条语句书写在一行上。语句书写完毕后,按 Enter 键转到下一行,开始下一条语句。有时,为了阅读方便,需要把一条语句写在几行上,这时就需要使用续行符了。PowerScript 的续行符是“&”,它放在一行的末尾,指示下一行是当前行的继续。例如,可以把语句:

```
MessageBox("SQL 语句出错","出错编号=" + String(SQLCA.SQLCode))
```

写在下面两行上:

```
MessageBox("SQL 语句出错","出错编号=" + &  
String(SQLCA.SQLCode))
```

注意,不能在标识符或保留字的中间续行。例如,对于上面的例子,下面的续行是错误的,原因在于将标识符分成了两行:

```
MessageBox("SQL 语句出错","出错编号=" + String(SQL&  
CA.SQLCode))
```

如果想在—行中写上多条语句,那么在语句之间用分号进行分隔,例如:

```
a = b + c; g = h + j; cnt = a + g + cnt
```

### 1.1.2 注释

注释是添加在程序中的一些说明性文字,PowerBuilder 并不执行任何注释,而是简单地跳过它。PowerScript 的注释有两种方式:行注释和块注释。如果熟悉 C 语言,就会发现 PowerScript 的注释方式与 C 语言完全相同。

行注释以“//”开始,其后书写注释内容,到行尾结束。例如:

```
// 这是一条行注释
```

```
Open(w_main) // 打开主窗口
```

块注释以“/\*”开始,到符号“\*/”结束,不管其中有多少东西,它们都是注释。例如:

```
/*
```

```
返回值的含义是:
```

```
0——出去
```

```
1——进来
```

```
*/
```

缺省情况下,在 PowerBuilder 的编程窗口中,所有注释均以蓝色显示。

注意,书写注释时不要使用续行符,使用多行注释符“/\* \*/”即可。同时,在嵌入式 SQL 语句中无需使用续行符,因为这类语句均以分号(;)作为语句的结束,完全可以把一条 SQL 语句写成多行。

### 1.1.3 标识符

标识符是程序中用来代表变量、标号、函数、窗口、菜单、控件、对象等名称的符号。在 PowerBuilder 中,标识符遵从下述规则:

- (1) 必须以字母或下划线开头。
- (2) 其余字符可以是字母、数字及下列特殊符号:下划线(\_)、短横线(-)、美元符号(\$)、号码符号(#)、百分号(%)。
- (3) 最长 40 个字符,中间不能插入空格。
- (4) 保留字不能用作标识符,因为它们已有特殊的含义。
- (5) 标识符不区分大小写,如 PART、Part、part 是同一个标识符。

例如,下面是一组正确的标识符:

```
rc                // 返回值
r_code           // 代码
Large_Button # 1 // 第一个大按钮
Child - ID       // 允许使用短横线时是个正确标识符
SpecialID        // 以下划线开头
```

下面的标识符写法是错误的:

```
Abc def          // 标识符中间不能有空格
abc > def        // 标识符中间有非法字符
```

This // 误用保留字

2th // 以数字开头

短横线与减号是同一个字符,因此,表达式中使用减法运算符时,必须在减号的两边加上两个空格,否则可能产生语法甚至语义错误。

注意,通常情况下,尽量不要在标识符中使用短横线(-),以免与减法运算混淆,用下划线分隔单词是较好的替代方法。如果想禁止在标识符中使用短横线(-),那么用文本编辑器打开 PB.INI 文件,在[pb]节中将 DashesInIdentifiers = 1 行修改为:

DashesInIdentifiers = 0

禁止在标识符中使用短横线(-)后,建立菜单时分隔条的名称也就不能使用短横线了。

另外,在 PowerBuilder 中,变量名和保留字也不区分大小写。

#### 1.1.4 保留字

保留字是 PowerBuilder 内部使用的一组单词,有着特殊的含义,不能再把它们用做标识符。如果应用程序误把保留字用做了标识符,那么 PowerBuilder 将给出编译错误。PowerBuilder 的保留字为:

|             |              |                 |            |               |                |
|-------------|--------------|-----------------|------------|---------------|----------------|
| alias       | and          | autoinstantiate | call       | case          | choose         |
| close       | commit       | connect         | constant   | continue      | create         |
| cursor      | declare      | delete          | describe   | descriptor    | destroy        |
| disconnect  | do           | dynamic         | else       | elseif        | end            |
| enumerated  | event        | execute         | exit       | external      | false          |
| fetch       | first        | for             | forward    | from          | function       |
| global      | goto         | halt            | if         | immediate     | indirect       |
| insert      | into         | intrinsic       | is         | last          | library        |
| loop        | next         | not             | of         | on            | open           |
| or          | parent       | post            | prepare    | prior         | private        |
| privateread | privatewrite | procedure       | protected  | protectedread | protectedwrite |
| prototypes  | public       | readonly        | ref        | return        | rollback       |
| rpcfunc     | select       | selectblob      | shared     | staic         | step           |
| subroutine  | super        | system          | systemread | systemwrite   | then           |
| this        | to           | trigger         | true       | type          | until          |
| update      | updateblob   | using           | variables  | while         | with           |
| within      | _ debug      |                 |            |               |                |

前面已经谈到,PowerBuilder 不区分标识符的大小写。与此类似,PowerBuilder 也不区分保留字的大小写。例如,IF、if、If、iF 指的都是同一个保留字,使用时可根据需要采用便于阅读的形式。一般情况下,代码中常把保留字书写为大写形式,关键在于同一个开发小组最好采用相同的约定。

#### 1.1.5 特殊字符

字符串中可以包括特殊的 ASCII 字符,它们不能使用常规的输入方法直接输入,需

要使用其他字符来代替。常用的特殊字符如表 1.1 所列。

表 1.1 常用的特殊字符

| 特殊字符               | 表示方法          |
|--------------------|---------------|
| 新行(Newline)        | ~ n           |
| 制表符(Tab)           | ~ t           |
| 垂直制表(Vertical Tab) | ~ v           |
| 回车(Carrige Return) | ~ r           |
| 换行(Formfeed)       | ~ f           |
| 退格(Backspace)      | ~ b           |
| 双引号(“ ”)           | ~ “ ”         |
| 单引号(‘ ’)           | ~ ‘ ’         |
| 波浪号(~)             | ~ ~           |
| 十进制形式的 ASCII 码     | ~ 000 到 ~ 255 |
| 十六进制形式的 ASCII 码    | ~ h00 到 ~ hFF |
| 八进制形式的 ASCII 码     | ~ 000 到 ~ 255 |

波浪号(~)用于引用特殊 ASCII 字符。

**示例 1** 下面的代码在字符串中使用了单引号(‘)：

```
string s1 = 'He said, "It ~ 's good!"'
```

**示例 2** 下面的代码在字符串中包含了回车换行字符,当该字符串显示在多行编辑框中时,将分成两行显示：

```
string s1 = "春眠不觉晓, ~ r ~ n 处处闻啼鸟"
```

### 1.1.6 空值

空值(NULL)是 PowerBuilder 与数据库交换数据时使用的一种特殊值,代表数据未定义、不确定,它与空字符串、数值零以及日期 00-00-00 的意义完全不同。

PowerBuilder 的所有数据类型都支持空值,但 PowerBuilder 并不将空值作为缺省值。例如,声明变量时,PowerBuilder 把 0 赋给数值型变量,把 False 赋给布尔型变量,把空串(“”)赋给字符串变量。

变量被赋予空值的途径有两条：

- (1) 从数据库中读到空值。
- (2) 使用 SetNull( )函数赋值。

例如：

```
String person      // person = ""
SetNull(person)    // person 值为 NULL
```

测试变量或表达式是否为空值时,使用函数 IsNull( ),而不是直接使用关系表达式。

例如,假设 a 是个变量,要测试它是否为空值,可以这样写：

IF IsNull(a) THEN ...

下面的写法是错误的:

IF a = NULL THEN ...

原因在于空值不等于任何值,也不等于另一个空值。

### 1.1.7 对象、属性、函数和事件的引用

PowerBuilder 应用程序的开发过程实际上就是各种对象的定义和使用过程。所有对象都有名称,并通过名称相互区分。在 PowerScript 中,访问对象的属性、函数、事件的方法很简单,就是使用圆点作为标记符。具体来说,访问对象属性的格式为:

对象名.对象属性

例如,把单行编辑框 sle\_name 的用户输入内容送到字符串变量 UserEnter 的语句可以写成:

UserEnter = sle\_name.Text

其中,sle\_name 是一个单行编辑框对象的名称,Text 是单行编辑框的 Text 属性。

程序中访问对象的函数与事件的格式为:

{ objectname. } { type } { calltype } { when } functionname ( { argumentlist } )

其中,大括号所括成分根据情况可以省略,各成分的意义为:

- (1) objectname 是对象名。
- (2) type 取值为 FUNCTION 或 EVENT,用于指明访问的是函数还是事件,缺省值为 FUNCTION。
- (3) calltype 用于指明 PowerBuilder 查找函数的时机,有效取值为:
  - STATIC(缺省值):编译时查找函数,若不存在,产生编译错误。
  - DYNAMIC:程序运行时查找函数,若不存在,产生运行错误。
- (4) when 用于指明函数或事件是立即执行还是当前程序段执行完毕后执行,其取值为:
  - TRIGGER(缺省值):立即执行。
  - POST:当前程序段执行完毕后执行。
- (5) functionname 指明调用的函数或事件名称。
- (6) argumentlist 给出函数或事件的参数。

例如,想把输入焦点移动到单行编辑框 sle\_name 上,程序中写上语句:

sle\_name.SetFocus( )

即可。如果想立即执行按钮 cb\_name 的单击事件处理程序,那么写上语句:

cb\_name.EVENT TRIGGER Clicked( )

即可。

### 1.1.8 This、Parent、ParentWindow 和 Super

PowerBuilder 提供了几个代词来指代特定的对象,它们分别是 This、Parent、ParentWindow 和 Super,其中前三个代词应用广泛。下面简单介绍一下这些代词的意义和用法。