

第4章 应用对象

教学提示：上一章学习了 Power Script 编程语言，本章将学习应用对象，主要介绍应用的基本概念、应用对象的创建、应用对象属性的设置、为应用对象添加事件脚本、脚本【PowerScript】画板的使用等有关内容。

教学要求：了解应用对象的有关概念；掌握创建应用对象的方法、应用对象属性的设置方法、为应用对象添加事件脚本、脚本【PowerScript】画板的使用、应用程序的运行。

4.1 创建应用对象

应用对象是 PowerBuilder 应用程序的入口和出口，一个应用程序的设计是从应用对象开始的。本节将介绍创建应用对象的方法。

4.1.1 应用的基本概念

下面，首先介绍有关应用的几个基本概念。

1) 应用

应用也称为应用程序。一般来讲，一个应用可完成一组相关的功能。

2) 应用的组成

在 PowerBuilder 中，应用是由一系列 PowerBuilder 对象所组成，这些对象包括：PowerBuilder 的应用对象、窗口对象、数据窗口对象、菜单对象、函数对象及用户对象等。

3) 应用的存储结构

在 PowerBuilder 中，应用以一定的层次结构存储，共有 4 个层次，即 Work Space(工作空间)、Target(目标)、PowerBuilder Library(应用库)、PB Objects(PB 对象)。

Work Space 是应用的第 1 层，文件以 .pbw 为扩展名。创建应用对象之前必须先创建工作空间。

Target 是应用的第 2 层，文件以 .pbt 为扩展名。一个工作空间中可以包含多个目标，创建应用对象的同时创建目标。

PowerBuilder Library 是应用的第 3 层，应用库文件以 .pbl 为扩展名。应用库是 PowerBuilder 提供的一种用于存储 PowerBuilder 对象的文件。一个完整的应用，既可以存放在一个应用库文件中，也可以存放在多个应用库文件中，即在一个目标中可以包含多个应用库。

PB 对象是应用的第 4 层。包括组成应用的所有对象，如应用对象、窗口对象、菜单对象等，其中必包含一个应用对象。

4) 应用对象

应用对象是 PowerBuilder 中一个非常重要的对象。应用对象没有可视化界面，只是提

供一些初始化、打开主窗口等操作的脚本。应用对象定义应用级属性和行为，如整个应用中的默认字体，应用开始、结束和出现系统错误时的操作等。

应用对象是进入应用的入口，所以说，创建应用的第一步是创建应用对象。下面介绍创建应用对象的方法。

4.1.2 创建新工作空间

在 PowerBuilder 9.0 中，创建应用对象之前必须先创建工作空间。创建工作空间的步骤与方法如下。

(1) 进入 PowerBuilder 后，选择【File】|【New】命令，或单击工具条的【New】图标，弹出新建对象对话框，选择【Workspace】标签页，如图 4.1 所示。

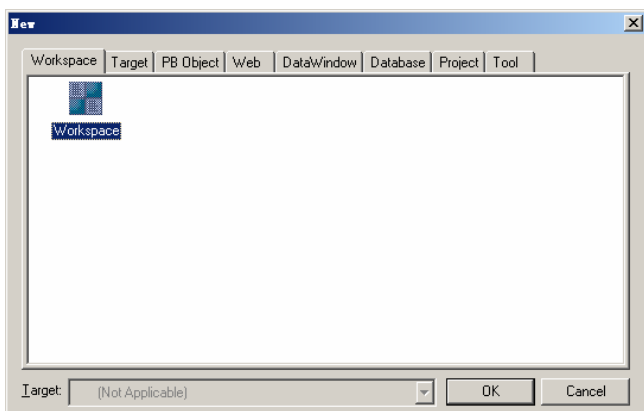


图 4.1 新建对象对话框【Workspace】标签页

(2) 在【Workspace】标签页中，选择【Workspace】图标双击，或单击【OK】按钮，弹出新建工作空间对话框，如图 4.2 所示。

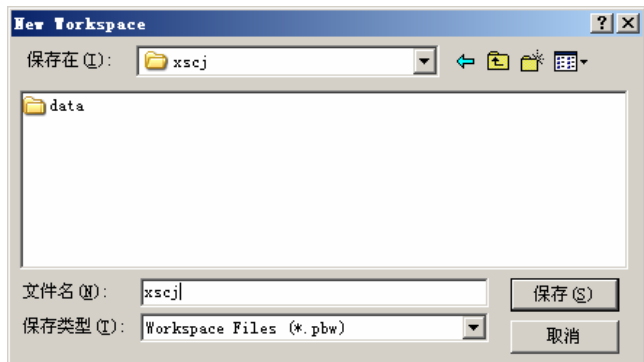


图 4.2 新建工作空间对话框

(3) 在【保存在】选项处选择路径，在【文件名】选项处输入文件名，【保存类型】选项处不做修改。本例中，工作空间名称输入为“xscj”，保存在 d:\xscj 之下。单击【保存】按钮，即可完成新工作空间的创建工作。

4.1.3 创建目标、应用库及应用对象

创建完一个新的工作空间后，就应该创建该工作空间的目标、应用库、应用对象，其具体步骤与方法如下。

(1) 选择【File】|【New】命令，或单击工具条中的【New】图标，弹出新建对话框，选择【Target】标签页，如图 4.3 所示。

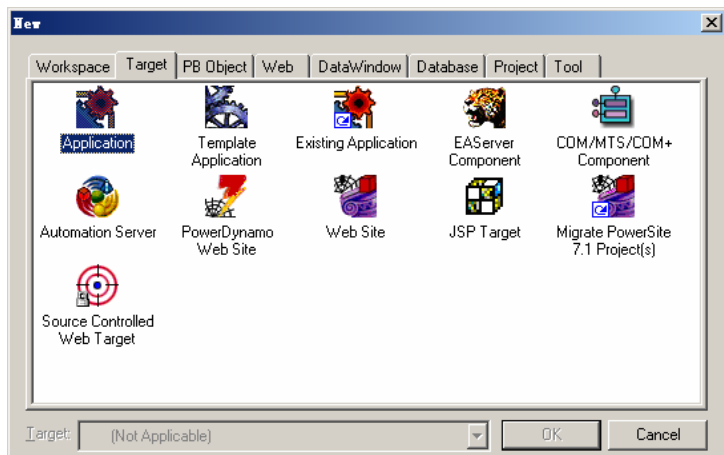


图 4.3 新建对象对话框【Target】标签页

(2) 在【Target】标签页中，选中【Application】图标双击，或单击【OK】按钮，弹出新建应用对象对话框，如图 4.4 所示。

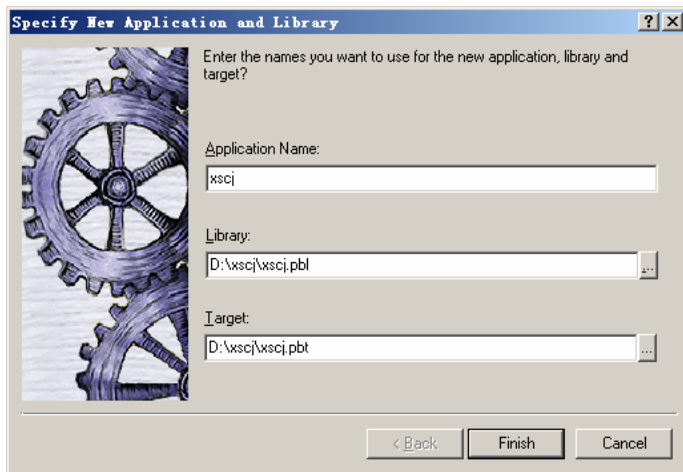


图 4.4 新建应用对象对话框

(3) 在【Application Name】文本框输入新建应用对象的名称，本例应用对象名称为“xscj”。在【Library】文本框输入新建应用对象所属库的名称，默认库名为应用对象名称。在【Target】文本框输入新建应用对象所属目标的名称，默认目标名称为应用对象名称。应用库、目标的路径均可以修改。填写完毕，单击【Finish】按钮，即完成了新应用对

象、目标、应用库的创建工作。

4.1.4 应用对象的存储结构

创建完一个应用对象后，在 PowerBuilder 主窗口的系统树窗口中显示出应用对象的层次存储结构，如图 4.5 所示。

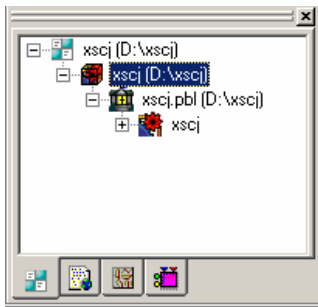


图 4.5 应用对象的存储结构

从图中可以清楚地看出应用的工作空间、目标、应用库和应用对象之间的层次结构关系。

需要说明以下内容。

(1) 一个工作空间可以包含多个目标。

如果要在当前工作空间中再新建一个目标，则应在该工作空间名上单击鼠标右键，在弹出的快捷菜单中选择【New】命令。在新建目标的同时，也新建了应用库和应用对象。

如果要在工作空间中新增一个已存在的目标，则在工作空间的快捷菜单中选择【Add Target】命令，在弹出的对话框中选择要加入的目标。

如果要在工作空间中删除某目标，则在工作空间的快捷菜单中选择【Remove Target】命令即可。删除该目标的同时，也从该工作空间中删除该目标下的应用库及其包含的对象。

(2) 一个目标中可以包含多个应用库。

应用库是在创建应用对象时一同创建的，只能通过修改目标属性的方法为目标增加或删除应用库文件。具体方法如下：

在系统树窗口中某个目标名上单击鼠标右键，在弹出的快捷菜单中选择【Properties】命令，弹出目标属性设置对话框，如图 4.6 所示。

在【Library Search Path】文本框中输入应用库的路径及名称，或单击【Browse】按钮选择应用库，将该应用库加到当前的目标中。这样目标就可包含多个应用库，各应用库文件以“；”号分隔。

如果要从目标中删除某个应用库，则选中该应用库文件，按键盘上的【Delete】键，即可删除该应用库。

(3) 一个完整的应用，既可以存放在一个应用库文件中，也可以存放在多个应用库文件中。

一个完整的应用只有一个入口和出口，所以如果一个完整的应用程序存放在多个应用库文件中时，只能有一个应用库文件中包含有应用对象。

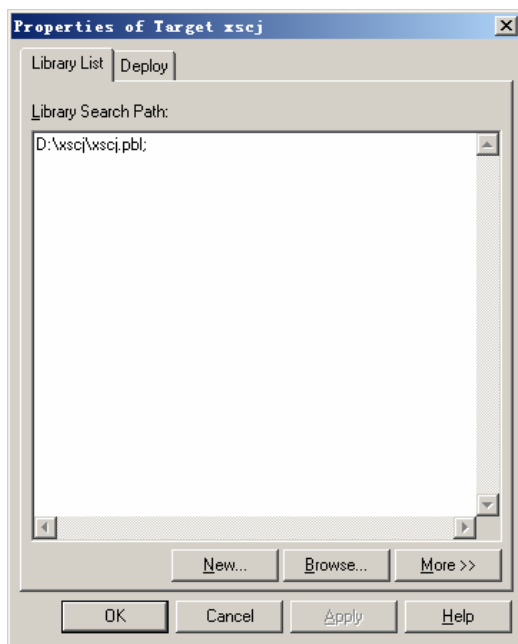


图 4.6 目标属性设置对话框

4.1.5 应用对象画板

双击系统树窗口中的应用对象图标，可以打开应用对象画板，如图 4.7 所示。

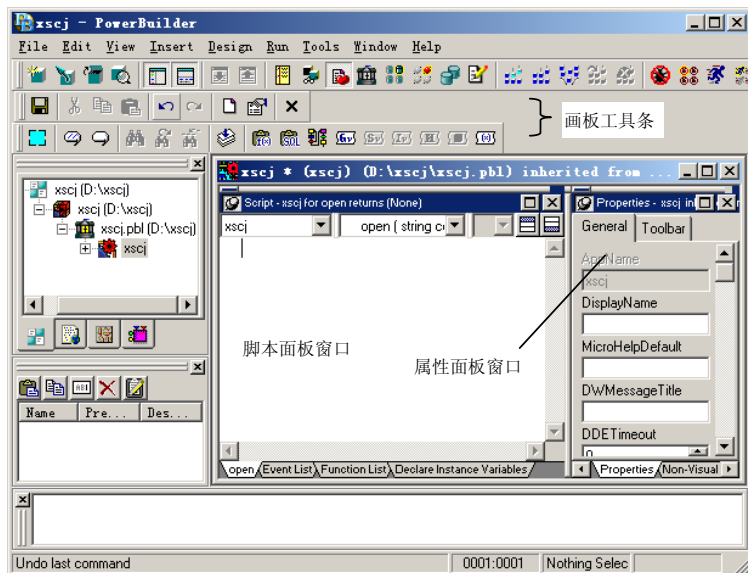


图 4.7 应用对象画板

新建应用对象画板窗口包括：脚本和属性两个面板窗口。

其中，脚本面板窗口包括：【Script】脚本(显示为 Open 事件)、【Event List】(事件列表)、【Function List】(函数列表)、【Declare Instance Variables】(定义实例变量)5 个视窗。

单击脚本面板下方的标签，可以选择不同的视窗。脚本视窗主要是为应用程序对象编写程序，【Event List】(事件列表)视窗主要是显示当前应用对象的所有事件，【Function List】(函数列表)视窗主要是显示当前应用对象的所有函数，【Declare Instance Variables】(定义实例变量)视窗主要是为应用对象添加变量。

属性面板窗口包括：【Properties】视窗和【Non-visual Object List】视窗。【Properties】视窗用于设置应用对象的属性，【Non-visual Object List】视窗是不可见对象的总列表。

打开应用对象画板窗口时，打开两个画板工具条，上面的工具条用于画板的一般操作。从左到右的按钮功能依次为：【Save】(保存)、【Cut】(剪切)、【Copy】(复制)、【Paste】(粘贴)、【Undo】(撤销)、【Redo】(恢复)、【Script】(脚本)、【Properties】(属性)和【Close】(关闭)。单击【Script】(脚本)按钮进入脚本视窗，可以编写事件脚本；单击【Properties】(属性)按钮进入属性视窗，可以设置应用对象的属性；其他按钮用于一般编辑操作。

4.1.6 运行应用

创建了应用对象后，还需要为应用对象设置属性和编写事件脚本，这些内容将在下面两节中详细介绍，然后就可以运行应用。当然在整个应用程序设计过程中随时可以运行程序，检查程序运行结果。

要运行程序时，单击主工具条上的【Run】图标即可。

运行出现错误时还可以进入调试状态对程序调试，调试方法将在第9章介绍。

4.2 设置应用对象的属性

本节将介绍如何设置应用对象的属性。

4.2.1 【Properties】窗口

【Properties】(应用对象属性设置)窗口在应用对象画板的右侧，如图4.7所示。

【Properties】窗口包含：【General】标签页和【Tool Bar】标签页，其中在【General】标签页中，存在一个应用对象附加属性窗口。

1. 【General】标签页

【General】(一般信息)标签页用于设置应用对象的显示名称、DDE通信时间等属性，如图4.8所示。一般只需设置【Display Name】选项，其默认值为应用程序名。

2. 【Tool Bar】标签页

【Tool Bar】(应用对象属性设置工具条)标签页用于设置应用对象的工具条属性，如图4.9所示。其主要属性及功能如表4-1所示。

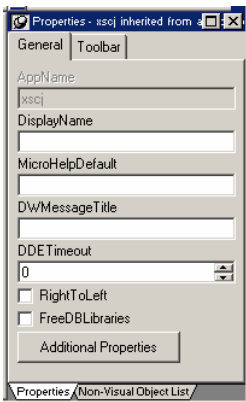


图 4.8 应用对象属性的【General】标签页

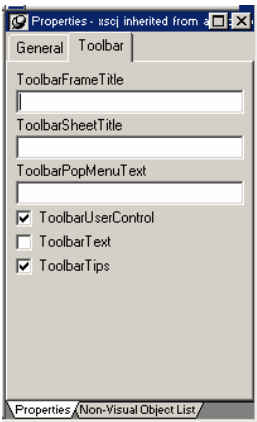


图 4.9 应用对象属性的【Tool bar】标签页

表 4-1 【Tool Bar】标签页的主要属性及功能

序号	属 性	功 能
1	Toolbar Frame Title	指定工具条的标题
2	Toolbar Pop Menu Text	指定工具条快捷菜单的位置，分 Left、Right、Top、Bottom、Floating5 种，各位置文本间用“，”隔开
3	Toolbar Sheet Title	指定当工具条处于浮动状态时的标题
4	Toolbar User Control	指定工具条快捷菜单是否可用
5	Toolbar Text	指定是否显示工具图标名称
6	Toolbar Tips	指定当鼠标移到该工具图标上时，是否显示该工具图标的提示信息

3. 应用对象附加属性窗口

在应用对象属性窗口的【General】标签页，单击【Additional Properties】按钮，弹出应用对象附加属性窗口，如图 4.10 所示。

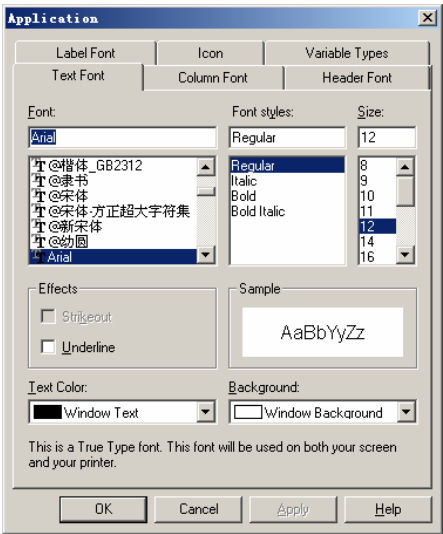


图 4.10 应用对象附加属性窗口

应用对象附加属性窗口包含：**【Text Font】**(文本字体)、**【Column Font】**(列字体)、**【Header Font】**(标题字体)、**【Label Font】**(标签字体)、**【Icon】**(图标)和**【Variable Types】**(变量字体) 6 个标签页。

4.2.2 设置应用对象的主要属性

1. 指定应用程序的图标

在应用对象附加属性窗口中选择**【Icon】**标签,如图 4.11 所示,输入复选框**【Icon Name】**的文件名,或单击**【Browse】**按钮选择复选框**【Icon Name】**的文件名,即可设置应用程序的图标。

一般应将应用图标文件设置成相对路径,以使编译后的应用程序具有可移植性。

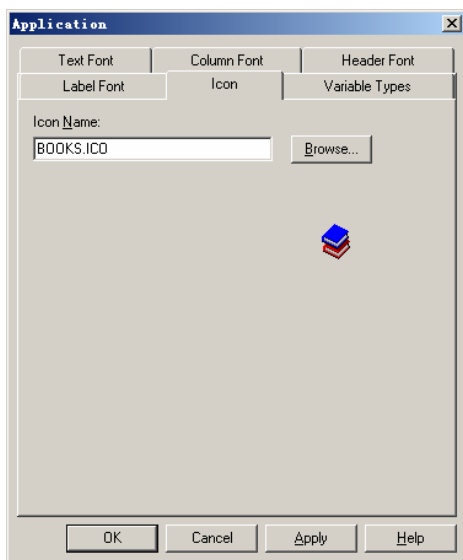


图 4.11 应用对象附加属性窗口的**【Icon】**标签页

2. 指定应用程序的字体

应用对象附加属性窗口中的**【Text Font】**(文本字体)、**【Column Font】**(列字体)、**【Header Font】**(标题字体)、**【Label Font】**(标签字体) 4 个标签页,用于设置整个应用程序中包含的对象的字体。开发应用程序时,设置应用对象字体可以保证整个应用程序中字体的 consistency。

1) **【Text Font】** 标签页

应用对象附加属性窗口的**【Text Font】**标签页,如图 4.10 所示,用于设置窗口中、用户对象和数据窗口对象中静态文本的默认字体。

2) **【Column Font】** 标签页

应用对象附加属性窗口的**【Column Font】**标签页用于设置在数据窗口对象中检索出的数据的默认字体。

3) **【Header Font】** 标签页

应用对象附加属性窗口的**【Header Font】**标签页用于设置在列表格式、网状格式的数据

据窗口对象中列标题栏的默认字体。

4) 【Label Font】标签页

应用对象附加属性窗口的【Label Font】标签页用于设置在自由格式的数据窗口对象中列标签的默认字体。

各标签页可分别设置：【Font】(字体的字形)、【Font Style】(风格)、【Size】(大小)、【Text Color】(颜色)、【Background】(背景颜色)、【Effects】(效果)等。

3. 设置应用程序默认的全局对象

应用对象附加属性窗口中的变量【Variable Types】标签页，如图 4.12 所示，可设置应用程序默认的全局对象。

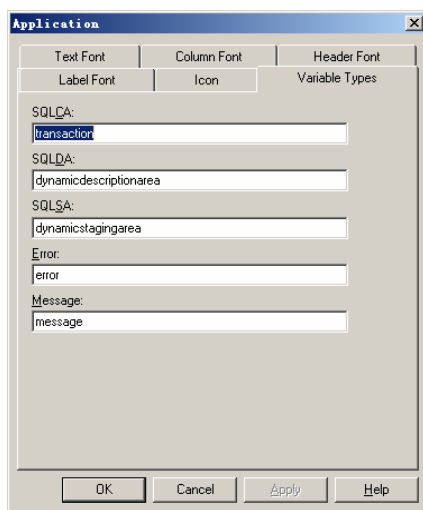


图 4.12 应用对象附加属性窗口的变量【Variable Types】标签页

全局对象是指其作用域为整个应用程序，也就是说，在整个应用程序的任何位置均可使用。图中所示的变量名是 PowerBuilder 为应用程序预定义的 5 个全局对象，其作用如表 4-2 所示。通常情况下，用户对这些系统变量名的类型不做修改。

表 4-2 PowerBuilder 预定义的 5 个全局对象

全局对象	用 途
SQLCA	transaction 对象，即事务对象，用来与数据库通信
SQLDA	dynamic description area，用于动态 SQL
SQLSA	dynamics tagging area，用于动态 SQL
Error	用于在执行过程中报告错误信息
Message	用于处理非 PowerBuilder 定义的事件的信息及对象间传递参数

4.3 为应用对象添加事件脚本

应用对象共有 6 个事件，分别为：Open 事件、Close 事件、System Error 事件、Idle 事

件、Connection Begin 事件、Connection End 事件。

4.3.1 应用的 Open 事件

应用对象的 Open 事件在打开应用时触发，其主要功能是实现与数据库的连接、初始化有关参数并且打开应用的主窗口，其典型的脚本如图 4.13 所示。

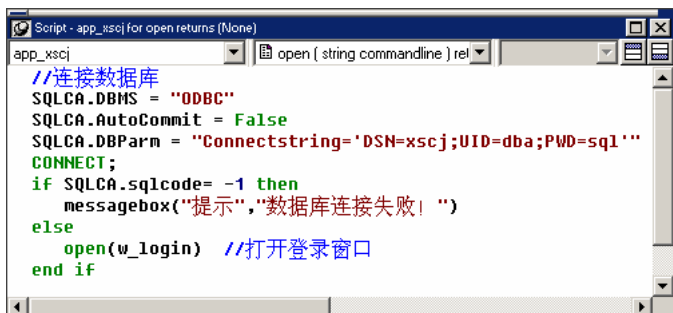


图 4.13 应用对象 Open 事件脚本

语句说明：

脚本的前 3 条语句为 SQLCA 的属性赋值，SQLCA 为应用的默认事务对象，用来实现前后台的数据通信。其中，参数 DBMS 指定数据库管理系统；参数 AutoCommit 指定自动提交属性；参数 DBParm 指定数据库连接参数，Connectstring 表示连接字符串，DSN 为数据源的名称，UID 为用户名，PWD 为用户密码。

第 4 条语句实现 SQLCA 与后台数据库的连接。

第 5 条语句是 If 结构语句，判断数据库连接是否成功。如果 SQLCODE 为 -1，则连接失败，利用 messagebox() 函数给出提示信息；否则，利用 Open() 函数打开主窗口。

4.3.2 应用的 Close 事件

应用对象的 Close 事件在关闭应用时触发，常用于断开与数据库的连接，如以下脚本：

```
destroy sqlca
```

语句说明：destroy 语句的功能是删除 SQLCA 事物对象，断开与后台数据库的连接，释放连接资源。

4.3.3 应用的 System Error 事件

应用对象的 System Error 事件在应用程序出现系统错误时触发，如果没有为此事件编写，PowerBuilder 显示带有 PowerBuilder 错误号和消息文本的消息框；如果有事件处理程序，则 PowerBuilder 执行该程序。例如：

```
HALT CLOSE
```

语句说明：将立即关闭该应用程序。

4.3.4 应用的 Idle 事件

应用对象的 Idle 事件，当函数 Idle 被调用，并且达到该函数指定的空闲时间时触发，主要功能是写一些屏幕保护程序或存盘程序。

例如，应用对象的 Open 事件中调用了 Idle(300)，Idle 事件中有以下语句：

```
HALT CLOSE
```

语句说明：若应用处于 300 秒钟时间内无操作状态，将立即停止该应用程序。在 Idle 事件处理程序中，HALT CLOSE 语句将关闭此应用程序。

4.4 使用【Script】视窗

PowerBuilder 编程就是编写事件脚本。PowerBuilder 的对象是在画板中设计的，各种对象画板都包含有【Script】(脚本)视窗，用于编写程序代码。

4.4.1 【Script】视窗

1. 进入【Script】视窗

要为一个对象编写脚本，首先要进入【Script】视窗，其步骤如下。

- (1) 在系统树窗口中选取需要编写其事件脚本的对象，如应用对象 xscj。
- (2) 双击该对象，或选择其快捷菜单中的【Edit】命令，则弹出该对象的画板。
- (3) 单击画板左侧面板的【Script】标签(实际显示为一个事件名)，或单击工具条中的【Script】按钮，即进入【Script】视窗，如图 4.7 所示的左侧为【Script】视窗。

2. 脚本编辑工具条

打开 PowerBuilder 对象的画板后，工具条中增加了若干个画板工具条，其中一个脚本编辑工具条。如图 4.14 所示，其图标功能从左至右依次如下说明。



图 4.14 脚本编辑工具条

- **【Select All】**(全选)按钮：可以选中所有编辑文本内容。
- **【Comment】**(注释)按钮：将选中的代码行或光标所在代码行变为注释。
- **【Uncomment】**(取消注释)按钮：将选中的行或光标所在行的注释取消。
- **【Find】**(查找)按钮：查找文本，可以打开一个对话框，在对话框中输入所要查找的内容。
- **【Find Next】**(查找下一处)按钮：该按钮，可以进行查找下一个符合要求的文本。
- **【Replace】**(替换)按钮：单击此按钮，可以进行文本的替换操作。
- **【Compile】**(编译)按钮：对脚本进行编译操作。
- **【Paste Function】**(粘贴函数)按钮：将函数粘贴到光标处。

- **【Paste SQL】**(粘贴 SQL)语句按钮：将 SQL 语句粘贴到光标处。
- **【Paste Statement】**(粘贴程序控制语句)按钮：将 PowerBuilder 语句粘贴到光标处。
- **【Paste Global】**(粘贴全局变量)按钮：可以将应用程序中的全局变量粘贴到光标处。
- **【Paste Shared】**(粘贴共享变量)按钮：可以将应用程序中的共享变量粘贴到光标处。
- **【Paste Instance】**(粘贴实例变量)按钮：可以将应用程序中的实例变量粘贴到光标处。
- **【Paste Window】**(粘贴窗口)按钮：用于粘贴窗口对象名称。
- **【Paste Object】**(粘贴对象)按钮：用于粘贴控件对象名称。
- **【Paste Argument】**(粘贴参数)按钮：用于粘贴参数。

4.4.2 粘贴函数

粘贴函数 **【Paste Function】** 的操作步骤如下。

(1) 进入脚本视窗，选择 **【Edit】 | 【Paste Special】 | 【Function】 | 【Built-in】** 命令，或单击工具条中的 **【Paste Function】** (粘贴函数)图标，并选择 **【Built-in】** 选项，画板上会出现 **【Paste Function】** 菜单的函数选择框，如图 4.15 所示。

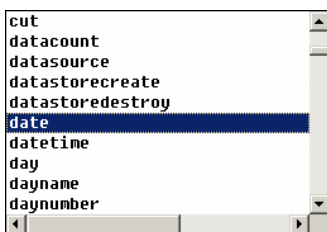


图 4.15 粘贴函数选择框

(2) 在函数选择框中选择要粘贴的函数双击，或单击后按回车键，此函数便粘贴到了脚本中。

注意：粘贴函数时，函数中没有一个参数被包含在内，用户必须自己填充函数的参数。

4.4.3 粘贴 SQL 语句

粘贴 SQL 语句时，系统自动和数据库连接，并进入到另外一个设计窗口，进行选择数据表、字段和条件等操作，最后将 SQL 语句返回，并粘贴到脚本画板的当前光标位置。

下面，以粘贴 Select 语句为例，说明粘贴 SQL 语句的操作过程。

(1) 选择 **【Edit】 | 【Paste Special】 | 【SQL】 | 【Select】** 命令，或单击工具条中的 **【Paste SQL】** 图标，并选择 **【Select】** 选项。

此时，系统会建立与数据库的连接。

(2) 建立数据库连接后，显示一个选择数据表的对话框，如图 4.16 所示。

用户根据需要提供所需的数据表(如选择“banji”)，选择完毕后，单击数据表列表框中的 **【Open】** 按钮，系统会打开插入值对话框，如图 4.17 所示。

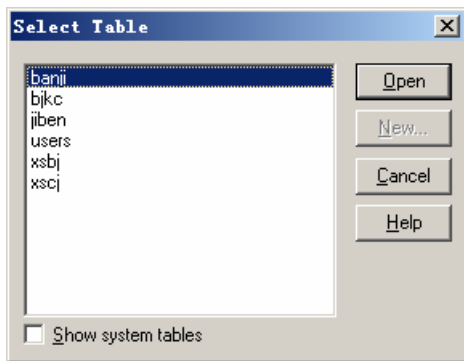


图 4.16 选择数据表对话框

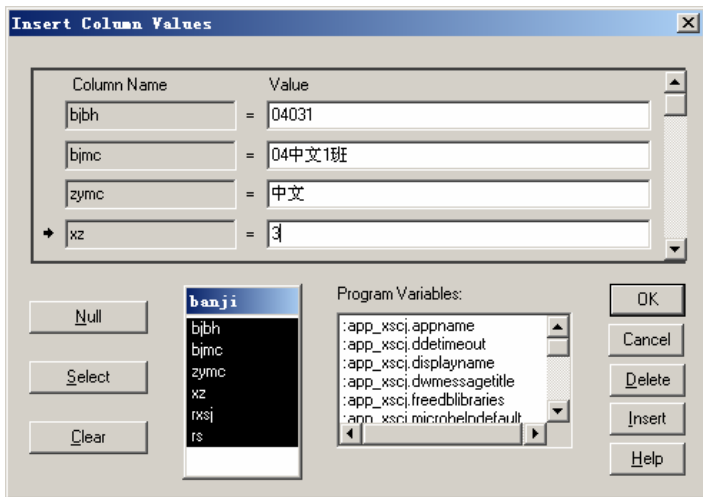


图 4.17 插入值对话框

(3) 在插入值对话框中,用户根据需要为每个字段输入插入值,输入完毕后,单击【OK】按钮,关闭此对话框,进入 Insert 设计窗口。

(4) 在 Insert 设计窗口中,用户可以进一步编辑 Insert 语句。关闭此窗口,则系统会弹出是否插入生成的 SQL 脚本语句的询问对话框,如图 4.18 所示。

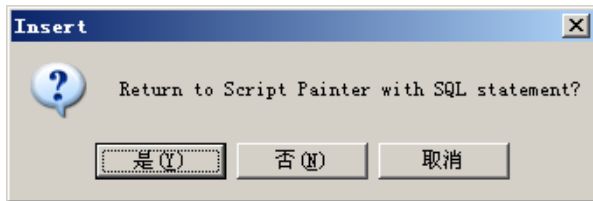


图 4.18 是否插入 SQL 脚本语句询问对话框

(5) 单击【是】按钮后,系统生成的 SQL 脚本语句被粘贴到脚本画板中光标的当前位置上,如图 4.19 所示。

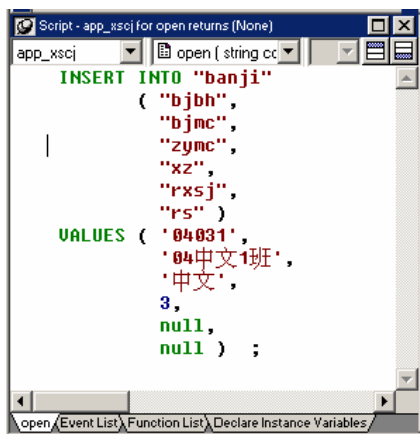


图 4.19 粘贴到脚本画板中的 SQL 语句

4.4.4 粘贴语句

要在脚本中粘贴一条 PowerScript 语句时使用【Paste Statement】(粘贴语句)功能键。下面以粘贴 If Then Elesif Else 语句为例说明其操作方法。

首先，在菜单栏中，依次选择【Edit】|【Paste Special】|【Statement】|【If Then Elesif Else】命令，或者单击工具条中的【Paste Statement】图标进行粘贴，并选择【If Then Elesif Else】项，则所选中的语句即可被粘贴到脚本中，如图 4.20 所示。

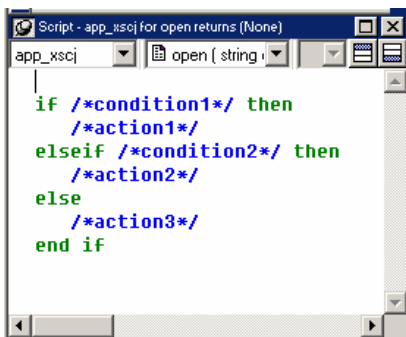


图 4.20 粘贴语句

然后，在脚本视窗中编辑此语句，如用实际的条件表达式替换其中的“/*condition1*/”等。

4.4.5 将脚本代码行设为注释

为了调试需要，经常把一些代码行暂时设为注释(即注释文本)，或设为注释的代码行还原为脚本代码(取消注释)。这两种情况，在脚本视窗中实现非常方便。

1) 【Comment】(注释文本)

首先，要注释连续多行时选中被注释的代码行，要注释单行时光标放在此行即可，然后单击工具条中的【Comment】图标，则将当前行或选中的多行变为注释行。注释行是以“//”开始的。

2) 【Uncomment】(取消注释)

首先,要取消连续多行注释时选中这些注释行,要取消单行注释时光标放在此行即可,然后单击工具条中的【Uncomment】图标,则将当前行或选中的多行变为代码行,注释引导符“//”消失。

提示: 如果你的脚本中存在错误,PowerBuilder 将不允许你离开脚本视窗。但有时,你可能需要离开脚本画板。此时,可单击工具条中的【Select All】图标,再单击【Comment】图标,可将整个脚本代码注释掉,就可以离开脚本画板。当你再返回时,可再次单击工具条中的【Select All】图标,并单击【Uncomment】图标,就可以还原为代码形式。

4.4.6 粘贴窗口和对象名称

对编程人员来讲,编辑代码经常要输入窗口或对象的名称,PowerBuilder 可以粘贴窗口或对象的名称,以方便操作和避免输入时的错误。

1) 粘贴窗口名称

选择【Edit】|【Paste Special】|【Window】命令,或单击工具条中的【Paste Window】(粘贴窗口)图标,会在光标处弹出窗口列表框,在该列表框中双击所要粘贴的窗口名称,即可将该窗口名称粘贴到脚本画板中。

2) 粘贴对象名称

选择【Edit】|【Paste Special】|【Paste Object】命令,或单击工具条中的【Paste Object】(粘贴用户对象)图标,会在光标处弹出对象列表框,在该列表框中双击所要粘贴的对象名称,可将该对象名称粘贴到脚本画板中。

4.4.7 导入文本文件

假定用户有保存着的文本文件程序代码,当前要编写的程序代码又与其相同或接近,这时可以导入已保存的文本文件程序代码,以简化程序代码的输入。

使用菜单命令可以导入文本文件,其操作过程如下。

选择【Edit】|【Paste Special】|【From File】命令,弹出选择文本文件对话框。将文件类型改为“All Files(*.*)”,然后找到所需要的文件并选中,如图 4.21 所示。

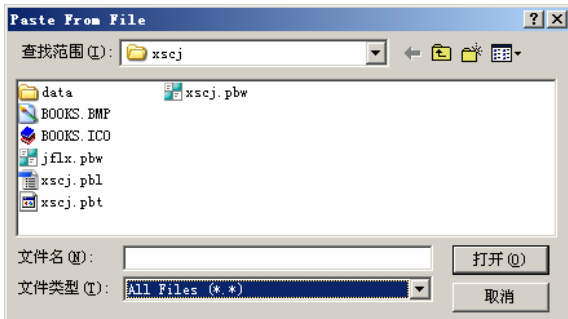


图 4.21 选择文本文件对话框

最后，单击【打开】按钮，可将该文本文件的内容导入到脚本画板中。

另外，工具条上还有其他一些粘贴按钮，使用非常简单。当要粘贴全局变量时，就选择【Paste Global】图标；粘贴共享变量选择【Paste Shared】图标；粘贴实例变量选择【Paste Instance】图标；粘贴当前事件参数选择【Paste Argument】图标。

4.5 小 结

本章介绍了应用的基本概念，重点讲述应用对象，包括：应用对象的创建、应用对象属性的设置、为应用对象添加事件脚本及使用【Script】视窗的技巧。

要求大家了解应用对象的有关概念；掌握创建应用对象的方法，应用对象属性的设置方法，为应用对象添加事件脚本；熟练掌握【Script】视窗的使用；并能上机操作练习，巩固以上所学内容。

4.6 实 训

实训目的

- (1) 掌握创建应用对象的方法。
- (2) 熟练掌握【Script】视窗的使用。
- (3) 掌握一个应用程序创建的具体操作步骤，并能运行该应用程序。

(4) 通过上机操作练习，巩固所学知识。包括：应用对象属性的设置方法，为应用对象添加事件脚本等所学内容。

实训内容

- (1) 创建一个应用程序。
- (2) 练习【Script】视窗的使用。
- (3) 设置应用对象属性。

实训步骤

- (1) 创建一个应用程序，并运行该程序。

工作空间命名为“sample.pbw”，应用对象名为“app_sample”，目标、库文件名称为默认值。

```
open 事件脚本:
time lt1
lt1=now ()
if lt1>time("07:0:0") and lt1<=time("12:0:0") then
    messagebox ("欢迎","早上好")
elseif lt1>time("12:0:0") and lt1<time("19:0:0") then
```



```
    messagebox ( "欢迎", "下午好" )  
else  
    messagebox ( "欢迎", "晚上好" )  
end if
```

(2) 练习【Script】视窗的使用：粘贴 now()、time()、messagebox()函数；粘贴 If Then Elseif Else 语句；给语句 If Then Elseif Else 添加注释，取消 If Then Elseif Else 语句的注释；导入文本文件等。

(3) 查看应用对象 sample 的存储结构。

(4) 设置应用对象 sample 的属性，将其字体改为宋体 9 号。

(5) 运行该程序。

4.7 习 题

1. 简述应用以及应用工作空间、目标、库文件、应用对象之间的关系。
2. 简述创建应用工作空间、目标、库文件、应用对象的方法。
3. 说明应用的存储层次结构。
4. 简述设置应用对象属性的方法，并详细说明为应用程序指定图标、为应用程序指定字体的具体步骤。
5. 应用对象的事件有几个？分别说明它们的作用。
6. 脚本编辑器应用工具条中有哪些功能图标，其作用是什么？
7. 说明粘贴函数，粘贴语句，粘贴窗口名称，粘贴对象名称，给语句添加注释、取消注释的操作过程。
8. 说明导入文本文件的操作步骤。
9. 说明运行应用程序的方法。