

第2章 数据库

教学提示：上一章我们已经学习了 PowerBuilder 9.0 的基本概念，本章我们将介绍数据库的相关知识，包括数据库文件的创建、数据库表的创建及数据的各种操作方法。

教学要求：通过本章的学习主要了解数据库基础知识，掌握数据库的设计步骤和数据库连接的方法；掌握数据库、表、主键、外键、索引、视图的作用与创建方法；掌握数据库数据的操作方式；掌握数据库的安全性的管理方法。

2.1 PowerBuilder 与数据库的接口

数据库操作是 PowerBuilder 的主要功能，利用 PowerBuilder 可以对各种各样的数据进行收集、加工处理及管理。那么在学习 PowerBuilder 的数据库管理操作之前，首先应对数据库的基础知识有所了解。

1. 数据、数据库、数据库管理系统、数据库系统

数据(Data)：数据是数据库中存储的基本对象。数据所包含的种类很多，文字、图形、图像、声音等，这些都是数据。可以对数据做如下定义：描述事物的符号记录称为数据。

数据库(Database，简称 DB)：顾名思义，数据库是存放数据的仓库。是指长期储存在计算机内的、有组织的、可共享的数据集合。数据库中的数据按一定的数据模型组织、描述和储存，具有较小的冗余度、较高的数据独立性和易扩展性，并可为各种用户共享。

数据库管理系统(DataBase Management System，简称 DBMS)：数据库管理系统是位于用户与操作系统之间的一层数据库管理软件。是数据库系统的一个重要组成部分。主要具备数据定义功能、数据操纵功能、数据库的运行管理、数据库的建立和维护功能。

数据库系统(DataBase System，简称 DBS)：是指在计算机系统中引入数据库后的系统，一般由数据库、数据库管理系统(及其开发工具)、应用系统、数据库管理员和用户构成。在一般不引起混淆的情况下常把数据库系统简称为数据库。

数据库一般由若干数据表组成，数据库中的数据就存放在数据表中。数据表一般由若干个记录组成，记录由若干个字段构成，字段主要是用来存放数据表中相同类型的一系列值。由此构成整个数据库也确定了它的大体结构，如图 2.1 所示。

2. PowerBuilder 与数据库的接口

应用程序要对数据库系统进行管理操作，必须先进行数据库的连接。而数据库的连接取决于开发程序与数据库的接口操作。用 PowerBuilder 开发的应用程序支持单个应用系统同时访问多个数据库系统，比如说，可以从 Oracle 中查询数据，然后将查询结果放到 Sybase 中，这样就实现了对两个数据库系统的同时访问。PowerBuilder 支持多种关系型数据库管理系统(RDBMS)，既包括 Oracle、Microsoft SQL Server、Sybase、Informix 等大型

数据库, 也包括 Foxpro、Xbase、Paradox 等个人数据库。对于大型数据库来说, PowerBuilder 提供了专用接口, 而小型数据库则通过 ODBC 接口灵活地进行访问。PowerBuilder 与其他数据库系统关系如图 2.2 所示。

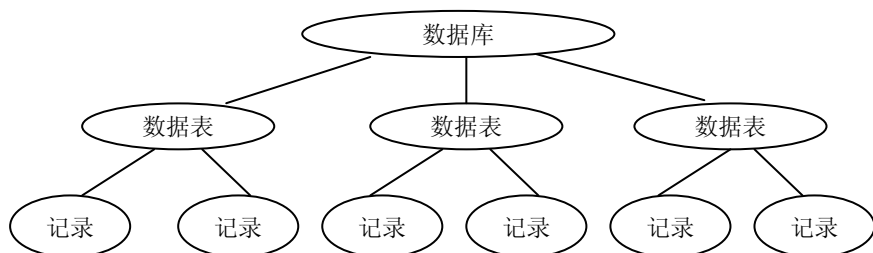


图 2.1 数据库结构示意图

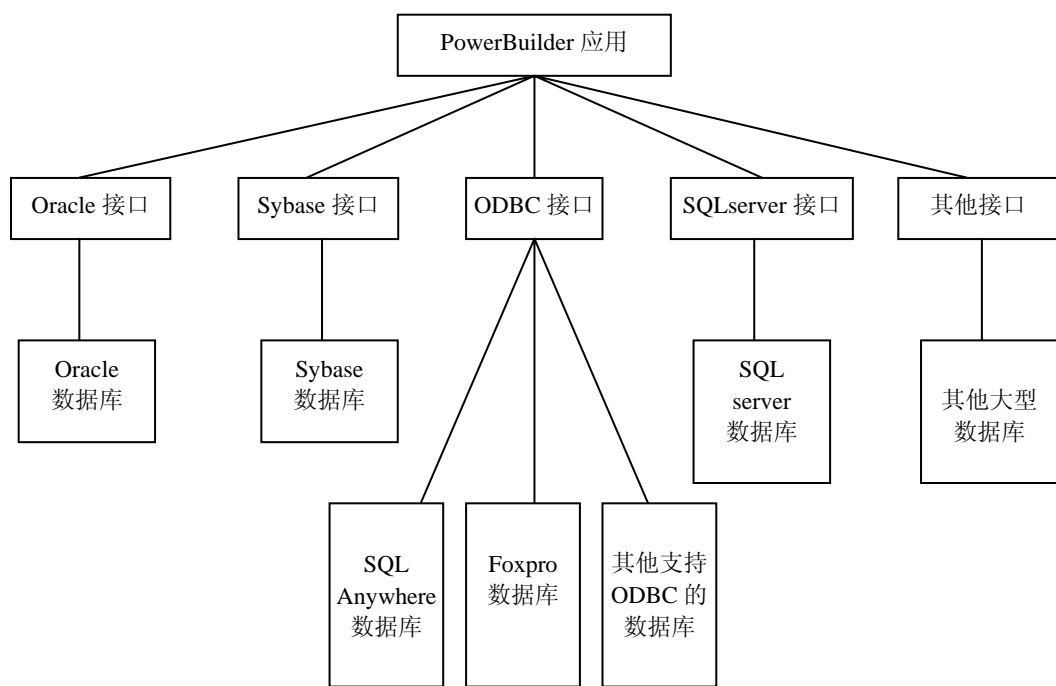


图 2.2 PowerBuilder 接口示意图

从图 2.2 可以看出, 数据库前端开发工具与后台数据库管理系统的连接方式是一个很重要的内容, 在连接时不仅要考虑到接口的专门性, 也要考虑到接口的通用性。PowerBuilder 针对这种情况提供了两种访问后台数据库的方式: 第一种是通过 ODBC 标准接口, 第二种是通过专用的接口与后台的数据库相连。ODBC(开放式数据库连接)是微软公司提出的数据库连接标准。使用 ODBC 方式连接数据库, 首先要做的就是创建数据源(具体操作我们在后面详细阐述), 创建好数据源后, 就可以在本机计算机中利用定义好的数据源存取后台数据库中的数据了; 还可以通过专用接口同后台的数据库相连。由于专用接口是针对特定的后台数据库管理系统而设计, 因此这种方式存取数据的速度要比采用 ODBC

方式存取数据的速度要快一些,而且更能够充分发挥特定数据库的优势。当然数据库、数据库的版本不同,使用的专用接口也不同。

PowerBuilder 9.0 是数据库应用系统的客户端开发工具,通过与数据库服务器的结合构成完整的客户机/服务器体系结构。在 PowerBuilder 能够与数据库管理系统建立连接之前,需要完成下述几方面的安装工作。

(1) 安装 PowerBuilder 9.0 本身。

(2) 安装数据库管理系统。

(3) 安装 PowerBuilder 9.0 与数据库的接口,Windows 2000 系统下由动态连接库(DLL)组成,一般是文件 `PB×××90.DLL`,其中 `×××` 是代表具体数据库的 3 个字符,例如,Microsoft SQL Server 专用接口的文件名为 `PBMSS90.DLL`。

(4) 安装数据库厂商提供的数据库接口文件,这些接口文件用于提供步骤(3)中的数据库接口访问数据库所需要的函数调用,例如,Microsoft SQL Server 的 32 位数据库接口文件为 `NTWDBLIB.DLL`。

(5) 安装数据库网络支持软件,这些软件与具体网络密切相关,变化极大。所以不能一概而论。一方面,PowerBuilder 向数据库应用程序提供了各种数据库的接口,使得我们的应用程序能够访问各种不同的异构数据库管理系统;另一方面,PowerBuilder 在集成开发环境中向开发人员提供了操纵和定义各种数据库的工具,也就是 PowerBuilder 的 DataBase Painter(数据库画板)。

另外,除了可以采用其他数据库管理软件创建数据库外,PowerBuilder 本身也带有一个功能完善的数据库 Adaptive Server Anywhere 8.0,可以直接在 PowerBuilder 内部方便地创建、管理数据库,为应用程序的数据库管理提供了工具。

2.2 创建 ASA 数据库

一般来说,在用 PowerBuilder 访问数据库前,我们使用相应的数据库管理系统(比如 Oracle, Sybase 等)创建自己的数据库(通常需要数据库管理员在服务器上完成创建工作),然后通过 ODBC 接口或专用接口把 PowerBuilder 9.0 或开发的应用程序连接到数据库上。对 FoxPro 之类的文件型桌面数据库来说,创建数据库只是建立一个存放数据库表的目录而已。PowerBuilder 本身提供的(ASA)数据库是一个完整的数据库管理系统,基于 Adaptive Server Anywhere 数据库开发的应用系统可以直接移植到网络环境下运行在其他数据库中。在这一节中我们主要讨论的就是如何在 PowerBuilder 中建立和删除本地 Sybase SQL Anywhere 数据库的方法。

2.2.1 使用 Database Painter 创建数据库

PowerBuilder 中,Database Painter(数据库画板)是专门用来管理与操作数据库的。主要包括创建和删除数据库、连接数据库、创建 ODBC 数据源、创建和管理表及操纵数据等功能。

1. 打开数据库画板

打开数据库画板的方法有两种，如图 2.3 所示。

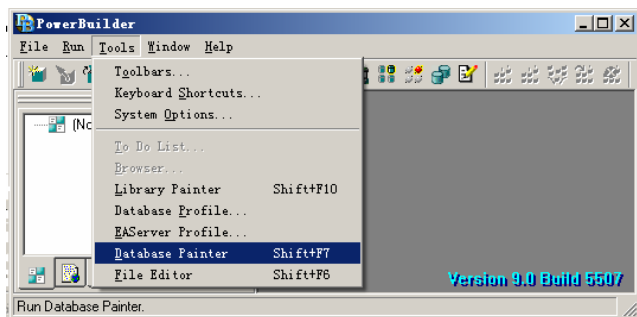


图 2.3 打开数据库画板

打开数据库画板后，系统显示如图 2.4 所示界面。

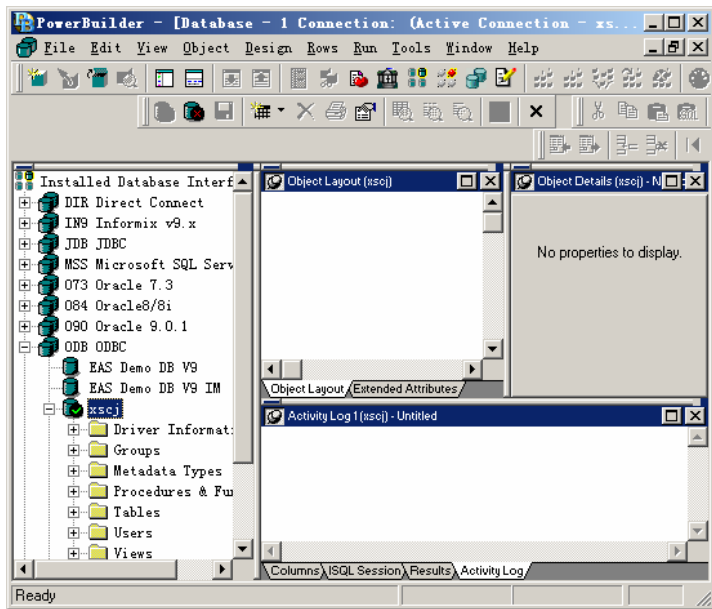


图 2.4 数据库画板显示视图

如图 2.4 所示，我们可以看到数据库画板工具栏和数据库画板工作区。数据库画板工作区中的对象视窗可以显示 PowerBuilder 支持的数据库接口，如 DIR Direct Connect、IN9 Informix v9.x、JDB JDBC、ODBC 及已创建的 DB profile、数据库文件，如“xscj”等。

在数据库画板工作区右边还包括对象布局视窗、对象细节视窗和列视窗，并可以显示相应的内容。

在数据库画板工作界面中可以利用数据库画板工具栏上提供的工具，如图 2.5 所示。

- (1) 创建、修改、删除表。
- (2) 创建、修改、删除视图。
- (3) 创建、删除索引。

- (4) 创建、修改、删除主键。
- (5) 创建、修改、删除外部键等操作。

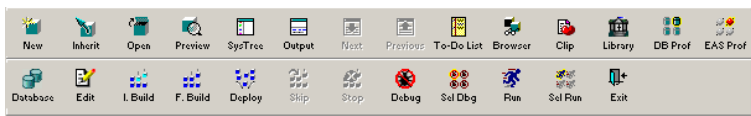


图 2.5 数据库画板工具栏视图

在上图中各工具功能如下。

- (1) DB Profile: 定义和修改数据库配置选项。
- (2) Open: 打开当前使用的相关文件。
- (3) Data manipulation(Grid): 以表格形式操作当前所选表中数据。
- (4) Data manipulation(Tabular): 以列表形式操作当前所选表中数据。
- (5) Data manipulation(Freeform): 以自由格式操作当前所选表中数据。
- (6) Pipeline: 定义和执行数据管道。
- (7) Close: 关闭数据库画板。

2. 创建 ASA 数据库

PowerBuilder 9.0 中提供了 Adaptive Server Anywhere 8.0(简称 ASA)数据库,可以快捷、方便地利用它进行数据库的设计。

进入数据库画板后如图 2.6 所示。

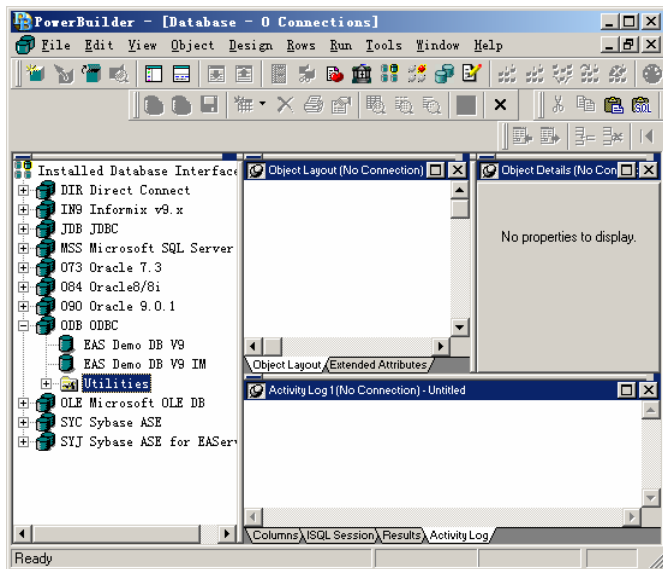


图 2.6 数据库画板初始视图

在如图 2.6 所示的对象视窗中,我们可以看到,由于没有建立新的数据库,所以图中只显示 PowerBuilder 中可以连接的数据库接口,关于数据库的具体内容是没有的。因此下面首先介绍如何在数据库画板中建立 ASA 数据库,如图 2.7 所示。

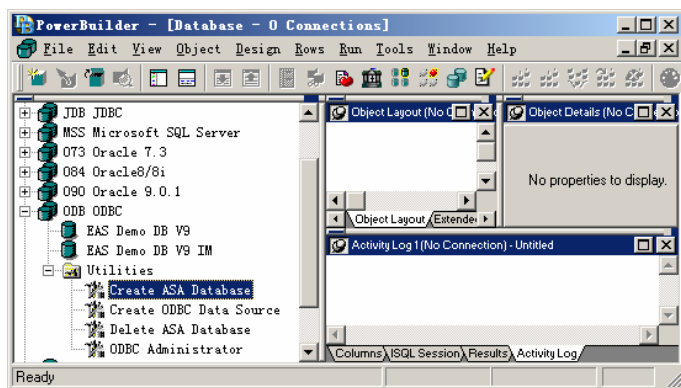


图 2.7 创建 ASA 数据库

展开如图 2.7 所示对象视窗中的【ODB ODBC】|【Utilities】|【Create ASA Database】命令，双击后出现对话框如图 2.8 所示。

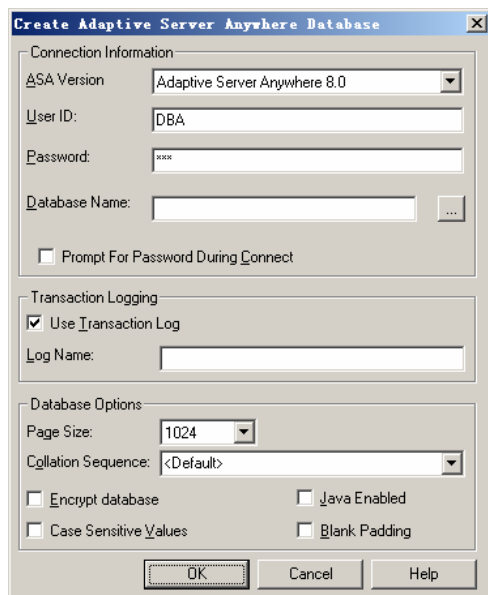


图 2.8 创建 xscj 数据库

其中选项如下。

(1) ASA Version: ASA 数据库版本。默认为 Adaptive Server Anywhere 8.0。

(2) User ID: 用户标识符。默认为 DBA。

(3) Password: 用户密码。默认为 SQL。

(4) Database Name: 要建立的数据库的名称。可以直接输入，也可以单击旁边的【Browser】按钮选择数据库的存放地点及名称，如图 2.9 所示。

(5) Database Options: 在这里可以设置数据库的一些属性。

保存后，在图 2.8 中单击【OK】按钮，便建立好如图 2.9 所示的 ASA 数据库“xscj”。

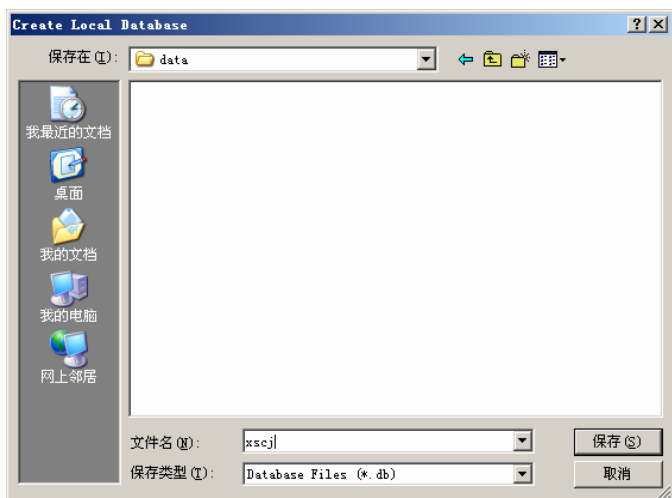


图 2.9 创建数据库名

2.2.2 使用 Sybase Central 创建数据库

一般安装 PowerBuilder 9.0, 同时会安装 Sybase Central 组件。使用 Sybase Central 组件可以进行数据库的创建、删除、压缩、解压缩、升级和备份等操作, 所以创建数据库也可以在 Sybase Central 这个数据库管理系统中进行。

建立步骤如下。

(1) 启动 Sybase Central。

选择【开始】|【程序】|【Sybase】|【Sybase Central】命令。弹出对话框如图 2.10 所示。

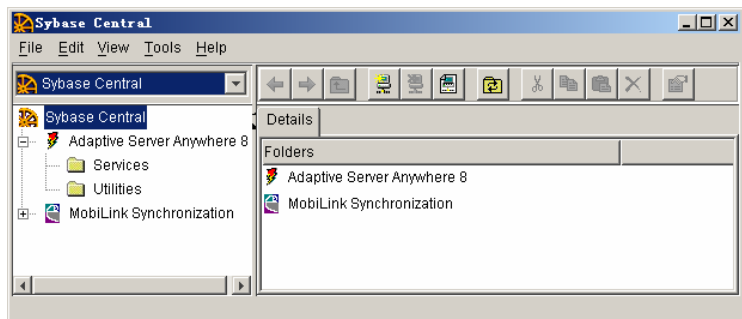


图 2.10 Sybase Central 操作界面

(2) 双击并打开【Utilities】选项, 然后双击【Create Database】选项, 进行数据库的创建, 弹出创建新数据库对话框, 如图 2.11 所示。在这里首先出现的是对向导的一个大概说明。可以单击【Next】按钮, 直接进行下一步操作或选中下方复选框, 以后不再显示此对话框。

(3) 如图 2.12 所示是创建本地数据库对话框, 我们选择第一个默认选项就可以建立本地数据库。单击【Next】按钮, 进入如图 2.12 所示的创建数据库对话框。

(4) 在如图 2.13 所示对话框中, 为新建的数据库文件指定存放路径与文件名。可以直接在编辑框中输入数据库文件路径与文件名, 也可单击【Browse】按钮进行选择操作。单

击【Next】按钮进入下一个对话框。

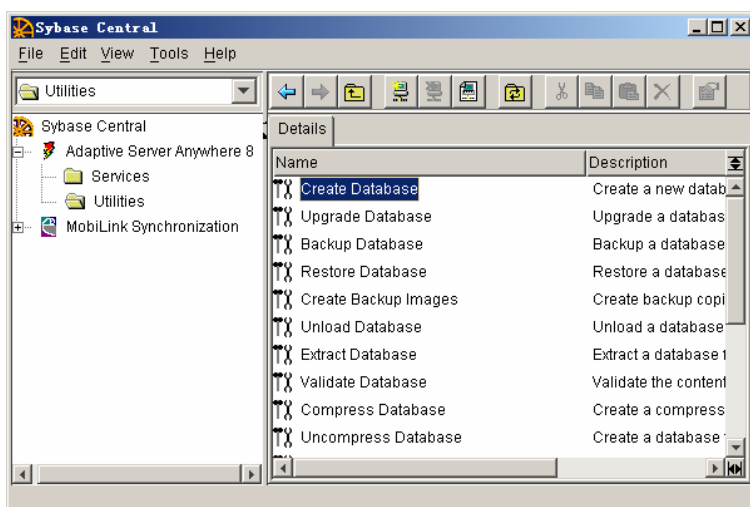


图 2.11 准备创建数据库

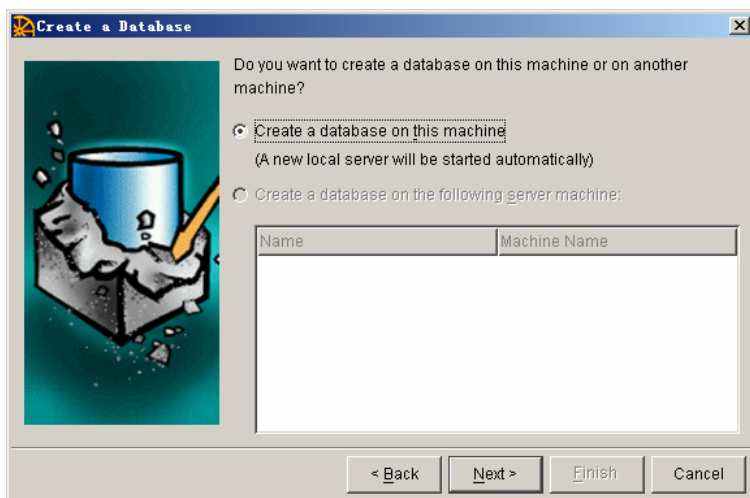


图 2.12 创建本地数据库对话框

(5) 在这里设置是否保存一个数据库日志文件。如选择【Maintain the following transaction log file】复选框, 则保存一个数据库日志文件并且文件名可默认为数据库文件名, 后缀为“log”。需要注意的是, 如果用户所选择的路径下有重名文件, 则该日志文件会将原文件覆盖。单击【Next】按钮进入下一个对话框。

(6) 在此对话框中, 用户可以选择是否要创建一个与日志文件同名的镜像文件。如选择【Maintain the following mirror log file】复选框则保存镜像文件。单击【Next】按钮进入下一个对话框。

(7) 在此对话框中, 用户可以选择是否支持 Java 类, 如果要求支持 Java 类就需要选中【Install Jconnect meta-information support】复选框。单击【Next】按钮进入下一个对话框。

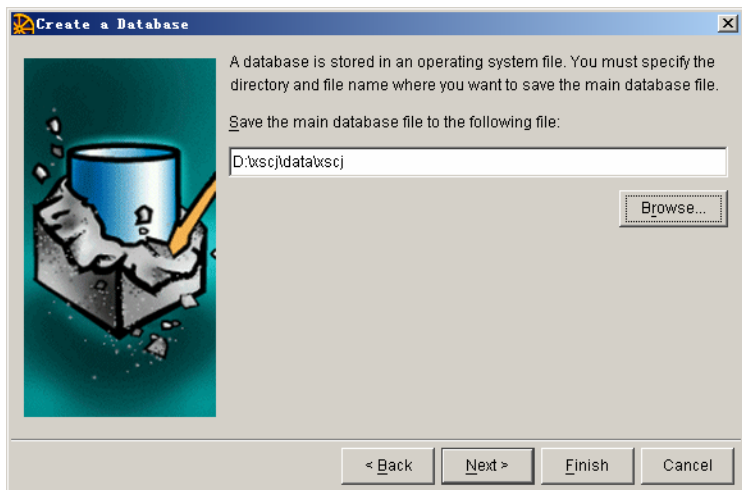


图 2.13 创建数据库文件名

(8) 在如图 2.14 所示对话框中，用户可对数据库进行设置。各选项功能如下。

- **Encrypt the database:** 对数据加密。
- **Ignore trailing blanks in comparisons:** 在比较两个字符串时忽略字符串尾部的空格。
- **Case sensitivity for all names and values:** 对所有的名和值都区分大小写。
- **Create SYSCOLUMNS and SYSINDEXES views:** 在创建数据库同时创建系统列视图和系统索引视图。

除了以上设置外，用户还可以单击对话框中的两个按钮进行不同的默认设置。单击【Next】按钮进入下一个对话框。

(9) 在此对话框中用户可以设置数据库页的大小。有几种数据库页大小模式可供用户选择。选择所要用的数据库大小模式后，单击【Next】按钮进入下一个对话框。

(10) 在此对话框可以设置排序依据。一般按默认排序依据。单击【Next】按钮进入如图 2.15 所示的对话框。

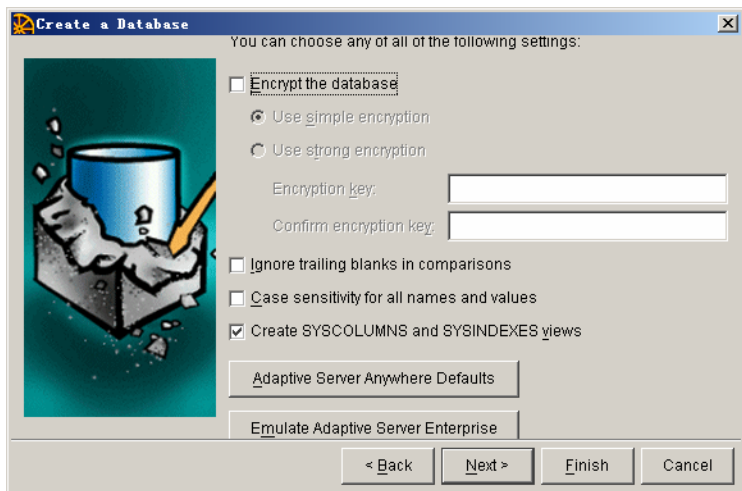


图 2.14 数据库属性设置

(11) 在如图 2.15 所示的对话框中选择【Connect to the new database】复选框就可以在创建数据库后自动进行连接。然后单击【Finish】按钮就可以进行数据库创建操作了。

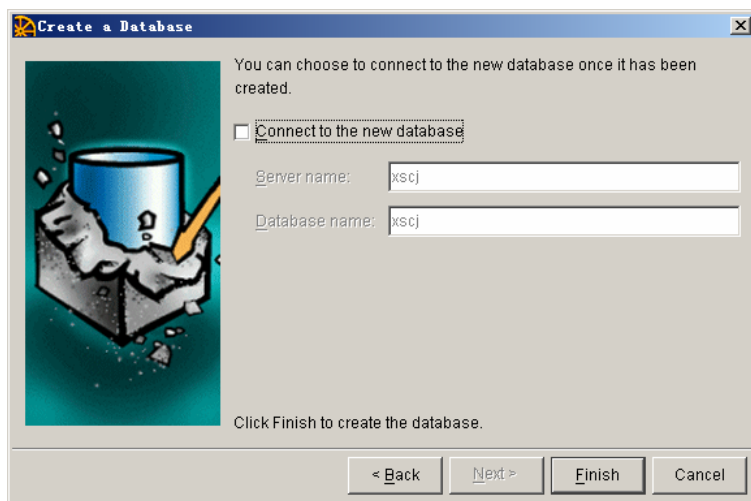


图 2.15 数据库连接

2.2.3 删除数据库

要删除数据库，只需在对象视窗中选择要删除的数据库名后右击，在弹出的快捷菜单中选择【Delete】命令，确认删除后就可以了。如图 2.16 所示。

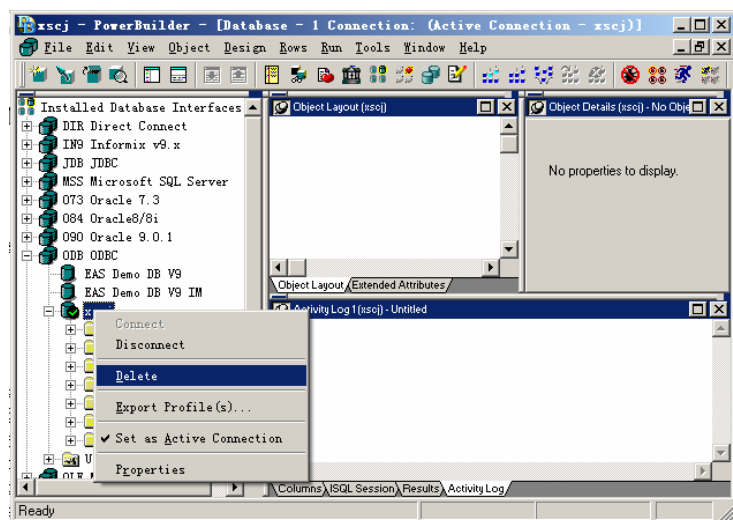


图 2.16 删除数据库

2.3 创建 ODBC 数据源和配置 Db Profile

前面我们介绍过 Powerbuilder 与数据库接口可以分为两种，一种是专用接口，主要针对一些专用的大型数据库如 SQL Server、Oracle 等；一种是 ODBC 标准接口。ODBC 的中文名字是开放式数据库连接，是微软公司提出的数据库连接标准。ODBC 数据源是很多数据库管理开发程序(如 VB, PowerBuilder 等)与数据库进行连接的接口之一，位于开发程序与数据库管理系统之间。开发程序可以通过 ODBC 接口连接到各种数据库管理的调用层。它主要由四层结构组成。

第一层为 ODBC 应用软件，包括用 PowerBuilder 开发的应用程序。应用程序通过调用 ODBC 函数，向 ODBC 数据源提供操作数据源的 SQL 语句，并处理 SQL 语句执行返回的结果。

第二层为驱动管理程序，或称作驱动管理器，它管理并装载驱动程序。驱动管理器的作用如下。

- (1) 用 ODBC.INI 文件把数据源名映射成特定驱动程序的动态链接库。
- (2) 处理几个 ODBC 初始化调用。
- (3) 为每个驱动程序提供 ODBC 的入口点。
- (4) 检验 ODBC 的调用参数和次序。

第三层为驱动程序，它处理 ODBC 函数调用，提交 SQL 请求到特定的数据源并把结果返回给应用程序，必要时它会修改应用程序的 SQL 语句，以适应特定数据库的 SQL 语法。

第四层是数据源，数据源包括要存取的数据以及相关的操作系统、数据库管理系统的网络系统。

我们已经在上一节介绍了创建 ASA 数据库的操作方法，在这种操作下系统会自动配置 ODBC 数据源和 DB Profile(描述文件)。但如果创建其他类型的数据库如 SQL Server、Sybase 和 Oracle 等，就需要经过两个步骤进行手动创建与配置：(1)创建和配置 ODBC 数据源及描述文件。(2)连接数据库。

2.3.1 创建 ODBC 数据源

不论是在 PowerBuilder 开发环境下操作数据库，还是在应用程序中使用数据库，如果是通过 ODBC 接口连接数据库，必须首先定义 ODBC 数据源。定义或创建数据源的方法一般有两种。

1. 通过 Windows 的 ODBC 配置工具创建 ODBC 数据源

在创建 ODBC 数据源之前，必须确保该数据源所对应的 DBMS 的驱动程序已经安装。

打开 Windows 控制面板，找到【ODBC 数据源工具】图标，双击打开如图 2.17 所示【ODBC 数据源管理器】对话框。



图 2.17 数据源管理器

2. 在 PowerBuilder 中利用【ODBC Administrator】工具

如图 2.18 所示。

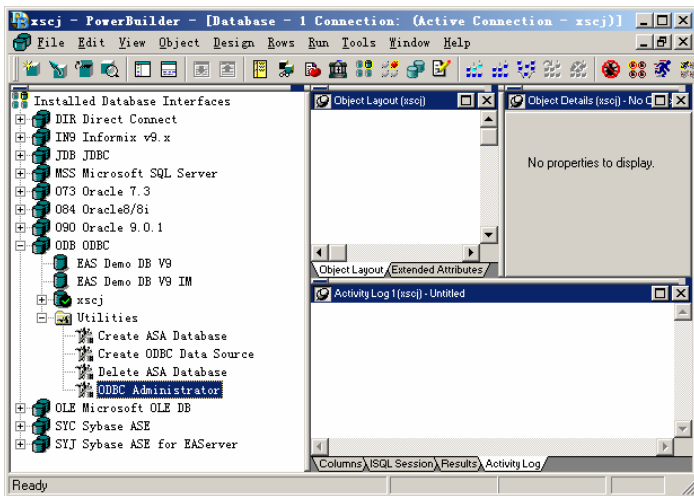


图 2.18 打开 ODBC Administrator

双击【ODBC Administrator】选项后，PowerBuilder 同样会弹出如图 2.17 所示的【ODBC 数据源管理器】对话框。该对话框中包含了 7 个标签页。

在【用户 DSN】、【系统 DSN】和【文件 DSN】标签页中，列出了当前可用的数据源。

(1) 用户 DSN 数据源对计算机来说是本地的，并且只能被当前用户访问，所以只有建立该数据源的用户才能访问，而且不能从网络上访问。

(2) 文件 DSN 可以在所有安装了相同驱动程序的用户中被共享，这些数据源不是用户专用或者对计算机来说不是本地的。

根据需要，可以选择不同类型的数据源。如果希望新建的数据只能供创建该 DSN 的登录用户使用，则选用【用户 DSN】标签；如果想让登录本机的所有用户都能使用，请选择【系统 DSN】标签；如果希望从网络上使用该数据源，则选择【文件 DSN】标签。

在【驱动程序】页面中，我们可以从中看到系统已经安装的数据库驱动程序。

下面我们以一个实例来介绍数据源的配置。

例如：我们要创建一个叫做“xscj”的数据源。

在这里要注意，我们要创建新数据源必须先创建一个 ASA 数据库。这种类型的数据库实际上就是一个扩展名为“.db”的文件，在这个文件中，ASA 不仅存放了数据库表中的数据，而且还存放了与这个数据库相关的配置信息。如果没有先建好这个库文件，是无法进行正常连接的。操作步骤如下。

(1) 在如图 2.17 所示的对话框中单击【添加】按钮，Windows 会弹出如图 2.19 所示的【创建新数据源】对话框。



图 2.19 创建新数据源

(2) 在如图 2.19 所示的对话框中，要求选择想安装的数据源驱动程序，在这里选择 Adaptive Server Anywhere 8.0，然后单击【完成】按钮。系统会弹出如图 2.20 所示的对话框。

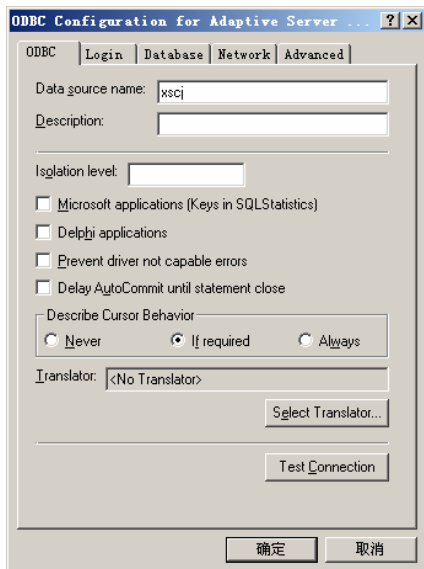


图 2.20 配置数据源

(3) 在这个对话框中共有 5 个标签页，其中在【ODBC】标签页中，我们需要在 Data source name 中给出新建数据源的名称：xscj。

(4) 在【Login】标签页中，需要提供连接数据库时要使用的用户名与密码。在这里在 User ID 处输入默认用户名“DBA”；在 Password 处输入默认密码“SQL”，如图 2.21 所示。

(5) 接着单击【Database】标签页，在这里需要提供所要连接的数据库名称及数据库文件所在的位置即文件路径，设定数据库名为“xscj”；数据库文件路径为“d:\xscj\data\xscj.db”，如图 2.22 所示(可以单击【Browse】按钮在计算机里选择)。同样要注意，此处的数据库文件必须是已存在的，否则将出现错误。

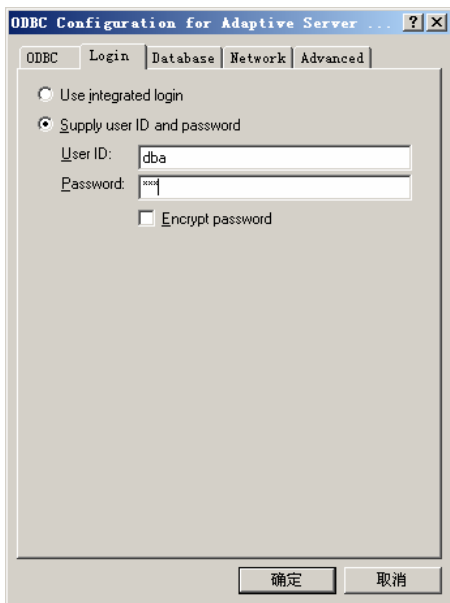


图 2.21 用户名及密码

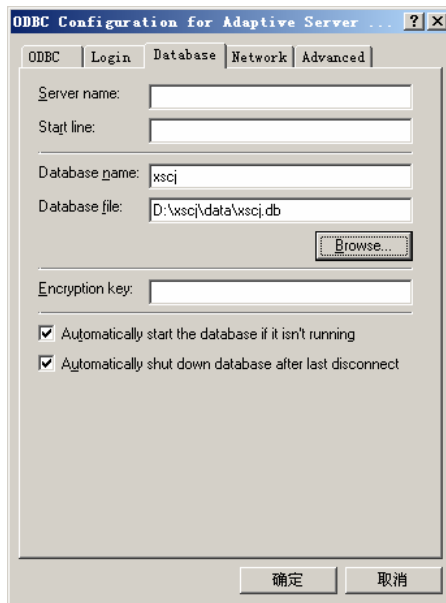


图 2.22 数据库设置

至此所需要设置的信息基本完成。现在回到如图 2.20 所示的【ODBC】标签页中。在下方有一个【Test Connection】按钮，可以通过这个按钮来测试配置的数据源是否成功。如果连接成功那么会出现如图 2.23 所示的提示框，表明连接成功；否则，系统会在弹出的提示框中提示出现错误的原因。



图 2.23 连接测试提示

2.3.2 配置 DB Profile

如果在 PowerBuilder 开发环境下操作数据库，只创建好数据源是不够的，还要建立

PowerBuilder 的数据库配置文件。所谓配置文件就是定义了一组连接到特定数据源或数据库的参数来指定要使用哪一个数据源。在 PowerBuilder 中,我们利用【DB Profile】工具来创建数据库描述文件。

配置 DB Profile 的方法如下。

(1) 单击画板工具栏上【DB Profile】图标,系统显示如图 2.24 所示窗口。

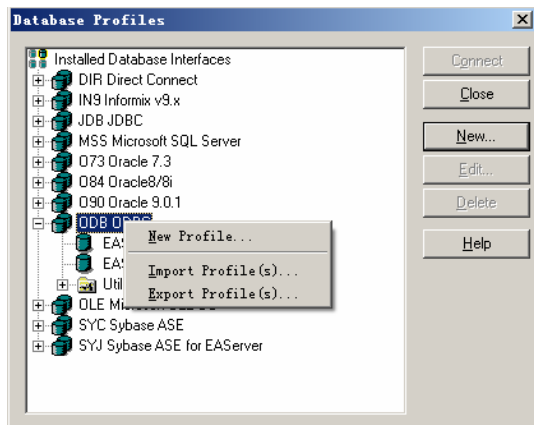


图 2.24 Database Profiles 对话框

(2) 在【ODBC ODBC】菜单上右击,选择【New Profile】子菜单;或者在窗口右边单击【New】按钮弹出如图 2.25 所示窗口。在【Connection】标签页中输入 Profile Name (配置文件名称)为“xscj”;Data Source(数据源)为“xscj”;User ID(用户名)为“DBA”;Password(密码)为“SQL”。

(3) 单击【OK】按钮,完成数据源配置。显示如图 2.26 所示。

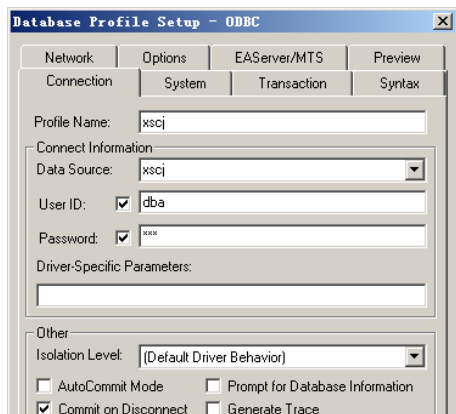


图 2.25 新建 DB Profile

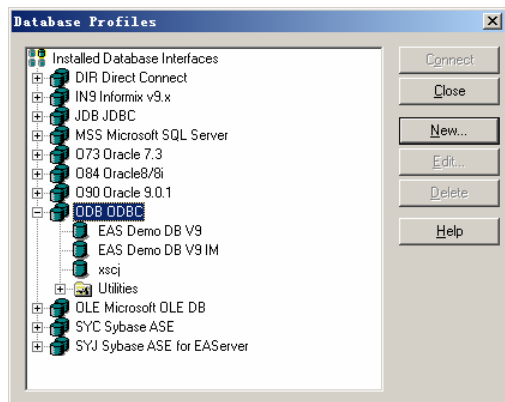


图 2.26 显示结果

完成以上操作就已创建好新的数据库配置文件。那么下一步需要做的就是通过【DB Profile】工具将开发环境连接到相应的数据源上。

2.4 连接数据库

在 PowerBuilder 开发环境与应用程序能够对数据库中的数据进行操纵之前，首先必须与数据库建立联系，也就是连接到数据库上。而方便有效地访问数据库是 PowerBuilder 的重要特色之一，下面就介绍在 PowerBuilder 环境中进行数据库连接的方法。

1. 连接本地 ASA 库

在如图 2.27 所示的窗口中，展开 ODB ODBC 前的“+”号，选择前面建立的数据库描述文件“xscj”。单击【Connect】按钮，系统连接到数据源定义的数据库上，如果连接过程出现错误，系统将给出相应提示。

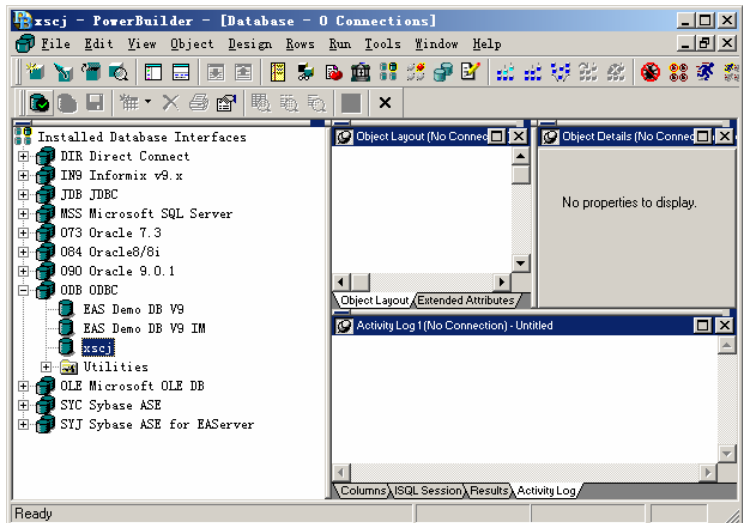


图 2.27 连接本地 ASA 数据库

2. 连接 SQL 库

使用专用接口可以大大地提高应用程序访问数据库的速度，所以应该尽可能使用专用接口而不是 ODBC 接口。下面以 MSS 数据库接口为例介绍专用接口的连接。MSS 数据库接口是 Microsoft SQL Server 数据库接口的简称，它支持的数据库管理系统是 Microsoft 的 SQL Server 数据库。

例如：事先用 Microsoft SQL Server 的企业管理器在 D:/my_database 下建立好数据库，文件名为“my_xscj.dbf”。

在如图 2.28 所示的对话框中填写相关信息，密码为“SQL”。单击【OK】按钮便可完成数据源配置，然后同上面操作相同，选中“xscj”数据源，单击【Connect】按钮完成 MSS 数据库连接。

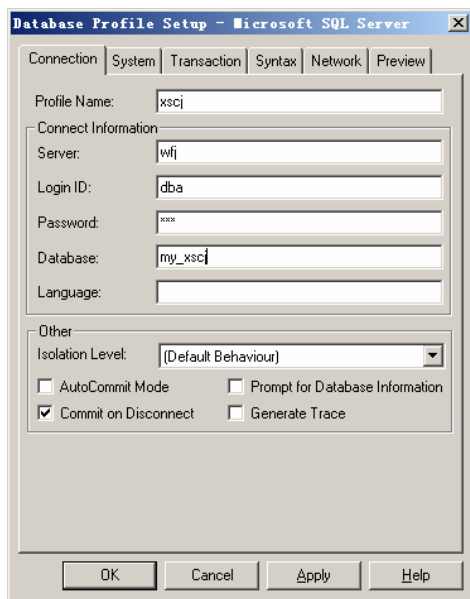


图 2.28 SQL Server 连接

2.5 创建数据库表

在数据库中，表是一个非常重要的组成部分，通过对数据库表的结构设置及数据的录入才能为数据库的各种操作与管理提供可能。数据库的表用行和列来表示数据，行就是记录，列就是字段，叫法虽然不同，本质上完全一样。操作数据库表包括几方面的含义：创建表、定义表中列的扩充属性、定义主键、创建索引、定义外部键、修改表、删除表、删除索引、删除主键。

2.5.1 创建数据库表

在使用数据库表之前必须首先创建数据库表。在 PowerBuilder 开发环境中，使用 PowerBuilder 的数据库画板创建新表有以下几种方法。

(1) 在 Database 画板的 Objects 子窗口中，选择【ODB ODBC】项，可以看到当前的数据库图标前有“+”号，单击它，展开后有一【Tables】选项，在【Tables】选项上右击，选择【New Table】命令，此时打开定义表结构窗口如图 2.29 所示。

(2) 从菜单创建表，可选择【Object】|【Insert】|【Table】命令来创建新表，如图 2.30 所示。

(3) 通过工具栏创建新表，操作步骤如下。

① 单击画板栏的数据库画板图标，进入数据库画板工作区，注意此处要求创建表的数据库应已连接成功，否则创建新表工具是不可以使用的。

② 单击数据库画板工具栏上的【Create Table】(创建表)图标如图 2.31 所示，进入表操作子窗口工作区，如图 2.32 所示。

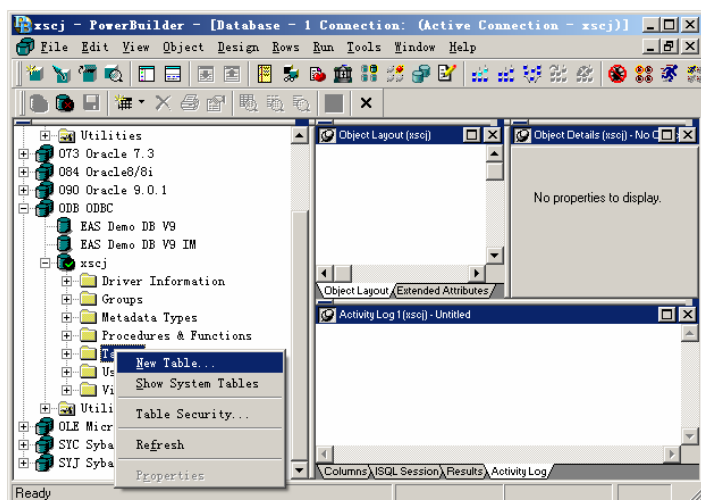


图 2.29 在对象区中新建数据库表

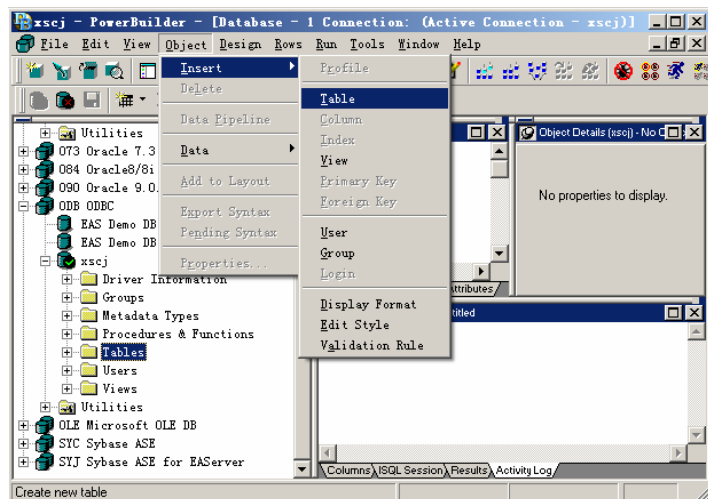


图 2.30 在菜单栏中新建数据库表

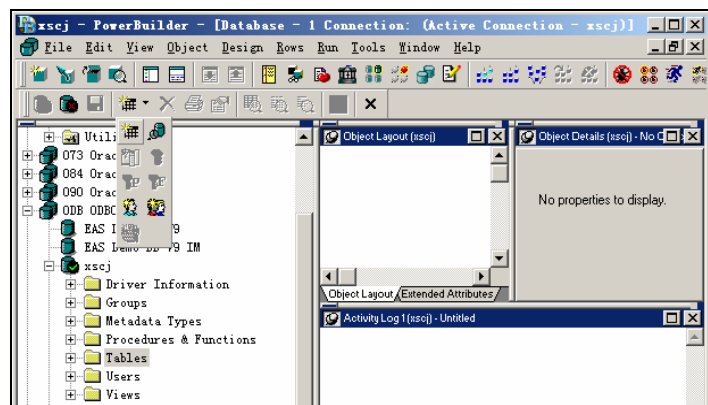


图 2.31 在工具栏中新建数据库表

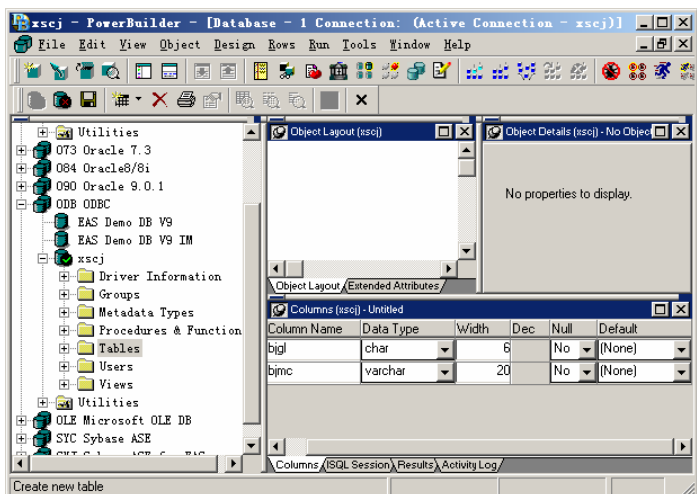


图 2.32 建立表结构

③ 键入表的各种属性，如：列名、列的数据类型、宽度、精度、是否允许为空等。如图 2.32 所示。

④ 单击工具栏上的【Save】按钮，保存表的定义。

注意：创建表时，以上几种方法的运用一定要注意：当前 PowerBuilder 要成功连接在指定数据库上，否则是不可以建立新数据库表的。

2.5.2 设置表和列的属性

在建立新表结构时要对表和列的属性进行设置，在图 2.32 中，我们可以看到每一行定义表的一个列的属性(输入时可以按 Tab 键将焦点移动到下一栏，按 Shift+Tab 键将输入焦点移动到上一栏)。其属性如下。

- **Column Name:** 表的列名。使用标识符作列名。
- **Data Type:** 指定列所存储的数据类型。PowerBuilder 以列表形式列出当前数据库支持的所有类型，可以直接从中选择所需类型。
- **Width:** 列的宽度。
- **Dec:** 小数位数。只有当列的数据类型选择为 decimal(或 numeric)时才需要输入这个值，它表示小数点后数字的位数。
- **Null:** 是否允许空值。Yes——允许，No——不允许。
- **Default:** 设置该列的缺省值。当用户向表中插入数据但没有在该列输入任何值时，系统就用缺省值填充该列。

其中 Null 是数据库中使用的一种特定值，它不同于零或空字符串，它表示“数据不确定”或尚未输入数据。

在图中定义好表的结构后，单击【Save】按钮出现如图 2.33 所示窗口，输入表名“banji”后，单击【OK】按钮，则新表创建完成。



图 2.33 保存表结构

下面以一个学生成绩管理系统为例，给大家介绍数据库表的建立操作。

(1) 首先确定建立的数据库表的结构。

数据库(data\xscj.db)共有 4 个表。

① 班级(banji)。

表 2-1 班级表列属性

列名	类型	宽度	null	解释
bjbh	char	6	No	班级编号
bjmc	varchar	20	No	班级名称
zymc	varchar	20	no	专业名称
xz	numeric	1.0	no	学制
rxsj	date		no	入学时间
rs	numeric	3.0	yes	人数

② 学生基本信息(jiben)。

表 2-2 学生基本信息表列属性

列名	类型	宽度	null	解释
xh	char	10	no	学号
xm	char	12	no	姓名
xb	char	2	no	性别
csrq	date		yes	出生日期
jtzz	varchar	20	yes	家庭住址
bjbh	char	6	yes	班级编号

③ 班级课程(bjkc)。

表 2-3 班级课程表列属性

列名	类型	宽度	null	解释
bjbh	char	6	no	班级编号
xq	char	9	no	学期
kcmc	varchar	20	no	课程名称
xs	numeric	3.0	no	学时
jsxm	char	12	yes	教师姓名

④ 学生成绩(xscj)。

表 2-4 学生成绩表列属性

列名	类型	宽度	null	解释
xh	char	10	no	学号
xq	char	9	no	学期
kcmc	varchar	20	no	课程名称
cj	numeric	5.1	yes	成绩

(2) 建立第一个表班级(banji)表。

用前面介绍的建立新表的一种办法,进入如图 2.32 所示操作窗口,设置新表的列属性,将相关内容输入如图 2.34 所示中。

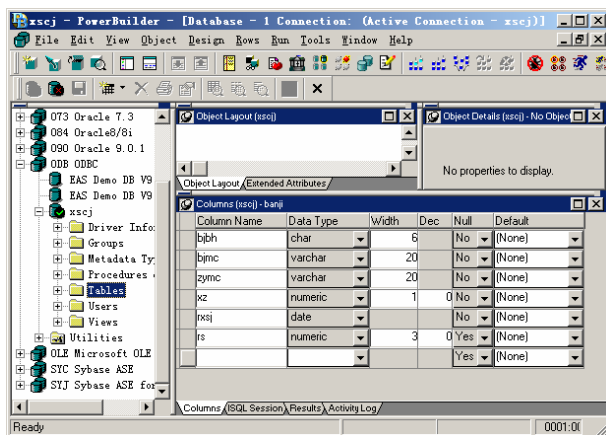


图 2.34 建立班级表

(3) 结构建立完毕后,单击工具栏上【Save】按钮,在弹出的对话框中输入班级表名称“banji”,如图 2.33 所示,完成表“banji”的建立。

(4) 然后采取同样的办法建立表:学生基本信息(jiben);班级课程(bjkc);学生成绩(xscj)。最后完成显示如图 2.35 所示。

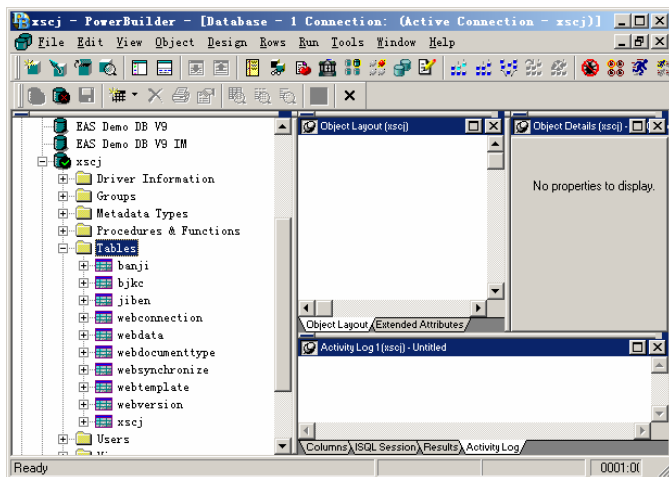


图 2.35 建立完毕

(5) 表的修改:表创建后,如果发现某些地方不符合要求,可以修改表的结构,但其中有些选项会被限制使用或不能用,例如,不能完成下列修改:改变列的类型、改变列的 Null 状态、缩小列的宽度、在表的中间添加新列。除此之外都可以进行修改。

2.5.3 创建主键、外键和索引

在数据库管理与操作中, 往往表与表之间都会存在这样或那样的关联: 可能某一个表会用到其他表的字段或内容, 或者在查询过程中某一个表的内容来自于其他表的查询结果; 在对表的记录进行排序时要以某一个字段进行排序。所以, 如何在数据库操作中完成以上的功能, 是在建立好数据库表后必须要做的事情。

首先了解以下几个概念。

主关键字: 用来唯一标识一条记录的一个或多个列的组合。在数据库中用主关键字来确保数据库的实体完整性。

外关键字: 是用于与另一个表进行联系的列。这里要求外关键字必须是另一个表的主关键字, 用来确保数据库的参照完整性。

索引: 主要目的是为了提提高数据库数据的查询速度。主要用于查询。

1. 建立主关键字

在建立主关键字时应当注意, 表中任意两行的主键键值都不相同, 如果存在相同键值, 则会产生重复行的错误。在选择作为主键的列时, 必须认真考虑、仔细设计, 既要使它能够唯一标识每一行, 又要使它不包括不必要的列。满足这些条件的选择方法可能有很多种, 但只能选择其一作为主键。

建立主关键字的方法有两种。

(1) 在展开的表名上右击, 选择【New】|【Primary Key】命令, 如图 2.36 所示。

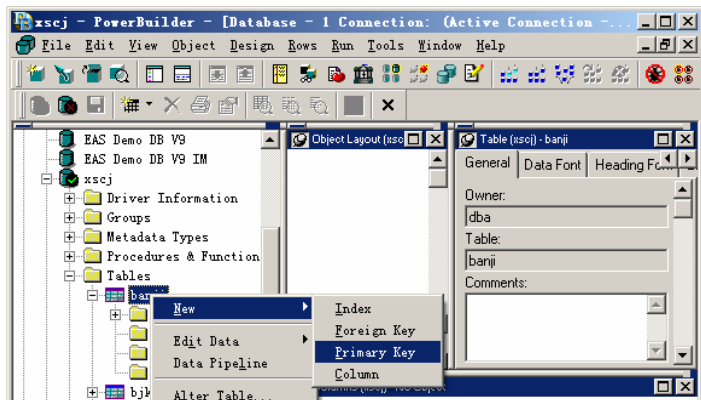


图 2.36 在表上创建主键

(2) 从数据库画板的 Painter 的下拉列表框中选择【Create new primary key for selected table】(建立主关键字)按钮, 如图 2.37 所示。

以上两种方法, 都会打开如图 2.38 所示对话框。在这里可以选择需要的列作为主关键字, 主关键字可以由一个列构成, 也可以是由若干个列构成的复合关键字。

以前面创建的学生成绩管理系统的数据库表中班级表(banji)为例, 在图 2.38 中选择“bjbh”为主关键字。选择完毕后, 单击工具栏上的【Save】按钮保存。然后依次为剩下的表设置主关键字, 分别为: 学生基本信息(jiben)设为“xh”; 班级课程(bjkc)设为“bjbh”,

“xq”，“kcmc”；学生成绩(xscj)设为“xh”，“xq”，“kcmc”。从中能够看出后面两个数据库表就是由多个列构成的复合关键字。

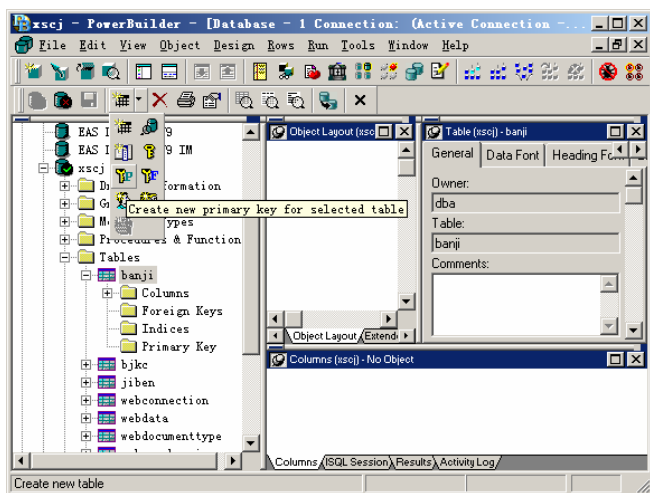


图 2.37 在工具栏上创建主键

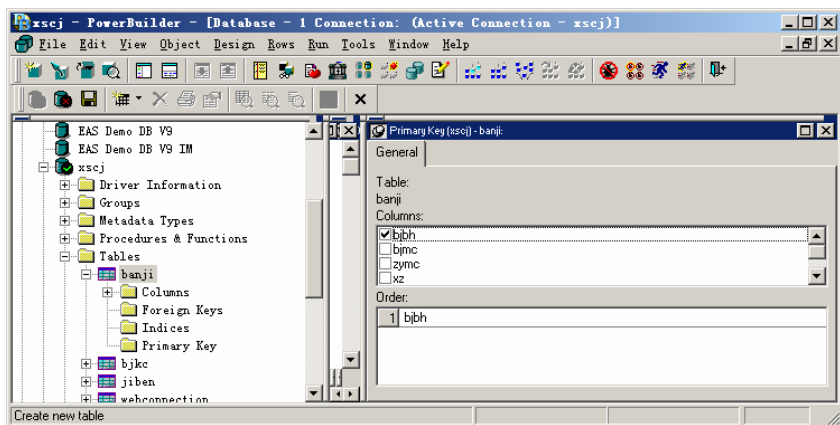


图 2.38 设置主键

2. 建立外关键字

外关键字主要是用来实现与其他表的关联，以确保数据的参照完整性。注意外关键字一定要关联表的主关键字。

建立外关键字与建立主关键字类似，也有两种方法。

(1) 在展开的表名上右击，在快捷菜单下选择【New】|【Foreign Key】菜单命令，如图 2.39 所示。

(2) 从数据库画板的下拉列表框中选择【Create new foreign key for selected table】(建立主关键字)按钮，如图 2.40 所示。

注意：在下面的例子中要建立“banji”表与“bjkc”表之间的关联。

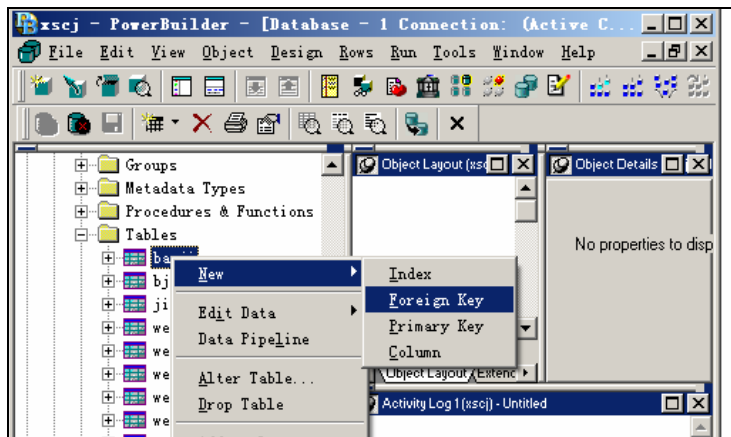


图 2.39 在表上创建外键

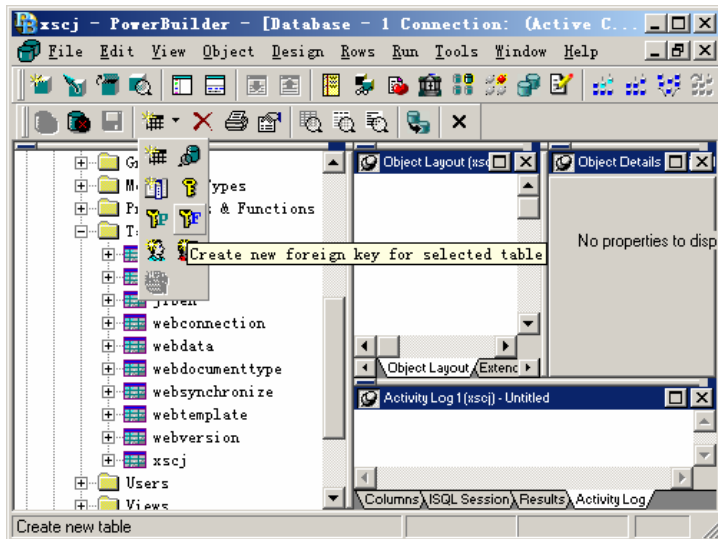


图 2.40 在工具栏中创建外键

以上两种方法都可以进入如图 2.41 所示的界面，其中有 3 个标签页，分别是：

①【General】标签页：其中，【Foreign Key】选项的功能是为外关键字定义一个名称。因为一个表可以有多个外关键字，为了区别与标识要给每个外关键字定义一个名称，在这里将【Foreign key】选项定义为“bjbh”。【Columns】选项的功能是选择构成外关键字的列，这里选择“bjbh”。

②【Primary Key】标签页：用来选择构成外关键字的列是哪个表的主关键字，如图 2.42 所示。

注意：【Table】选项的功能是用来选择与之构成关联的主关键字所在的表，必须要正确选择才能构成正确的关联关系。在这里选择“banji”。同样【Columns】选项的功能是选择表中主关键字，如果所选取的表已建立好主关键字，则此处将自动显示，当然在有必要的情况下可以手动修改。

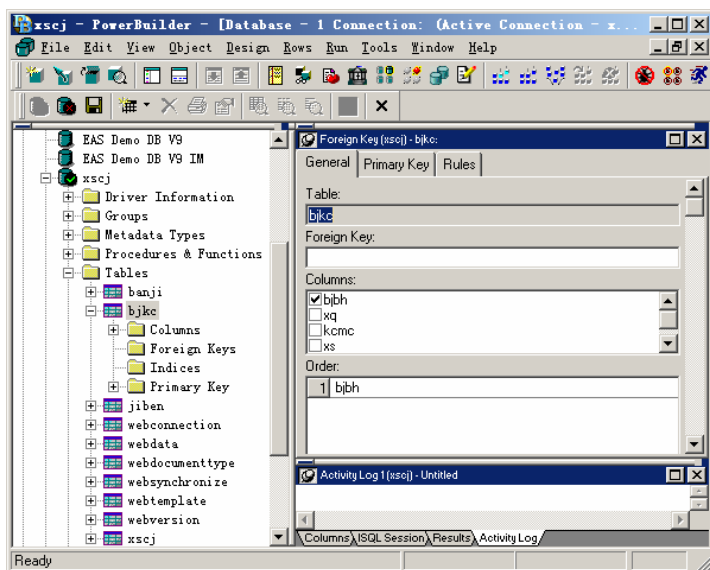


图 2.41 【General】标签页设置

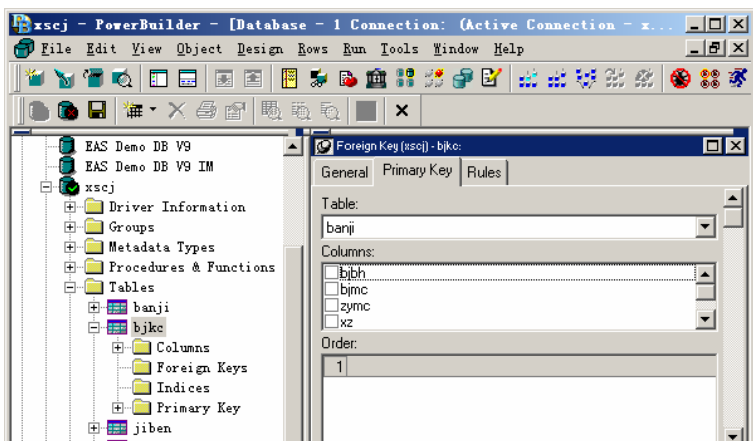


图 2.42 【Primary Key】标签页设置

③【Rules】标签页：这里选择删除主表的行时，要遵循的完整性规则如下。

- ◆ Disallow if Dependent Rows Exist(RESTRICT)：如果有依赖的子记录存在，则禁止删除主表中的记录。
- ◆ Delete any Dependent Rows(CASCADE)：一同删除所有依赖的子记录。
- ◆ Set Dependent Columns to Null(SET NULL)：将依赖的子记录的相应字段的值置为空值(NULL)。

注意：默认的约束规则为：如果有依赖的子记录，则禁止删除主表中的记录。在使用中一般选取这一选项。如图 2.43 所示。

当以上各项完毕后，单击工具栏上的【Save】按钮保存。

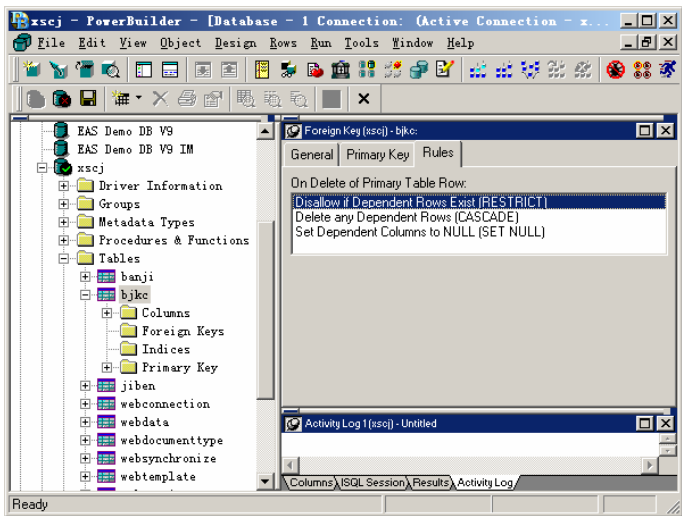


图 2.43 【Rules】标签页设置

3. 建立索引

在数据库操作与管理中,为了提高数据检索效率,通常我们都要对经常查询的列(字段)定义索引。索引是在数据库实现表中数据逻辑排序的方法。建立索引后,表中原本无序的数据可以按照设定的顺序进行输出。可以认为索引是指向表中各行数据的指针数组,这个数组按索引列进行排序。建立索引的列可以是一个列,也可以是多个列的组合。索引可以是重复索引,即允许数据重复;也可以是唯一索引,即不允许数据重复。通过索引能够方便地在数据库中查找到所需的数据。

建立索引有以下两种方法。

(1) 在展开的表名上右击,在快捷菜单下选择【New】|【Index】命令,如图 2.44 所示。

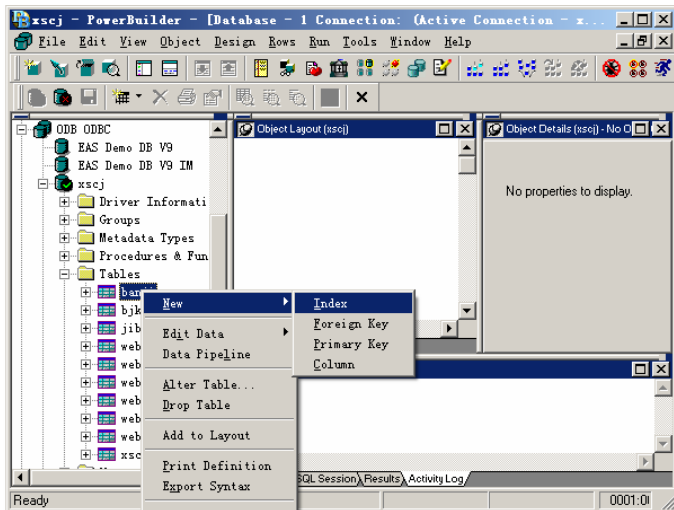


图 2.44 在表上创建索引

(2) 从数据库画板的下拉列表框中选择【Create new Index for selected table】(建立索引)

按钮,如图 2.45 所示。

以上两种方法都会进入如图 2.46 所示操作界面。

在图 2.46 中的各选项含义如下。

- **Index(索引名):** 由于一个表可以建立多个索引。为避免混淆和冲突必须为每一个索引定义一个名称,本例中定为 kcmc。
- **Columns(建立索引的列):** 选择建立索引的列。可以是一列,也可以是多个列的组合,本例中定为 kcmc。

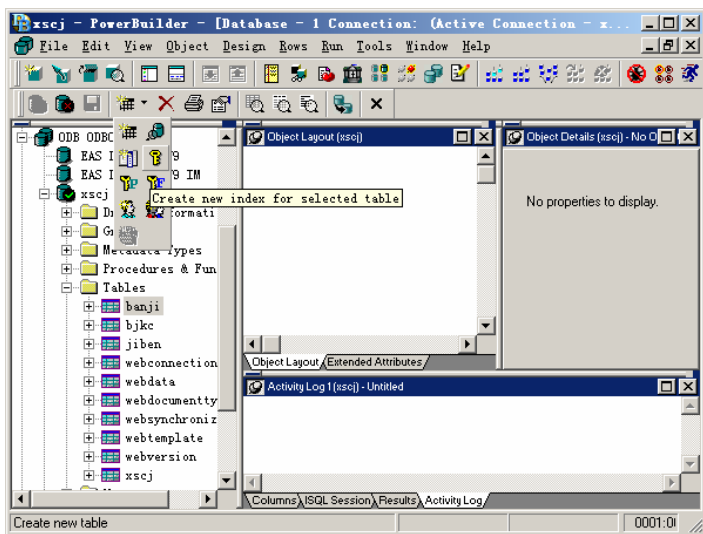


图 2.45 在工具栏上创建索引

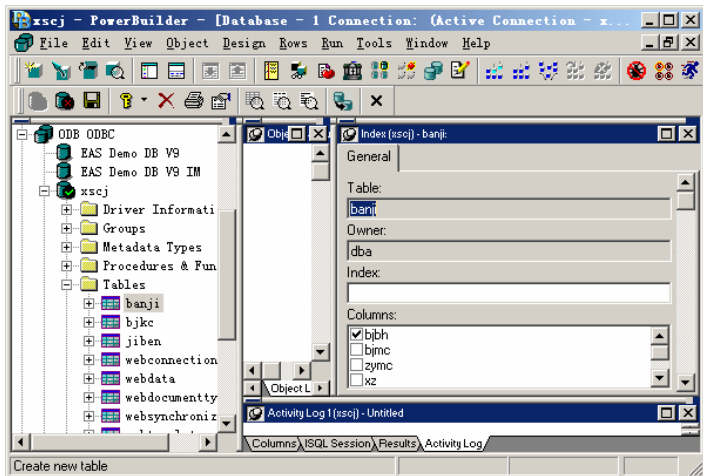


图 2.46 设置索引

- **Order(索引顺序):** 显示出所建立索引的顺序。即先按哪个列排,后按哪个列排。
- **Unique(索引唯一性):** 确定建立的索引是否是唯一性索引。选中则表示建立唯一性索引。
- **Ascending(升序排列):** 确定索引是否按升序排列。如是按升序排列则选中,否则将

按降序排列。

将所有选项内容确定好后，按【Save】按钮保存。

2.6 操纵数据

当数据库与数据表建立完毕后，我们可以利用 PowerBuilder 9.0 中提供的数据操纵工具对数据进行操纵，来完成数据的浏览、编辑修改、排序及过滤等操作。

在 PowerBuilder 9.0 中使用的操纵工具是 Data Manipulation，如图 2.47 所示。



图 2.47 操纵数据

Data Manipulation 画板主要包括以下 3 个工具按钮，分别对应 3 种显示风格，如图 2.48 所示。

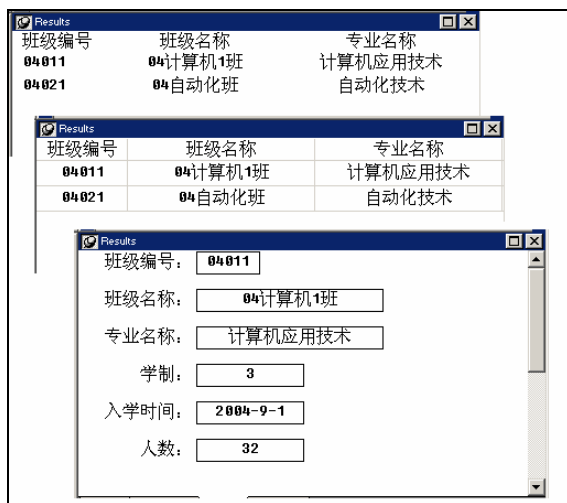


图 2.48 数据显示风格

- Freeform: 自由显示风格
- Grid: 网格显示风格
- Tabular: 行列显示风格

数据显示后,对数据的相应操作可以通过单击工具栏上的按钮来实现。工具栏上的按钮有如下几个:

- Retrieve: 从数据库中检索数据
- Insert Row: 向表中插入一行
- Delete Row: 从表中删除一行
- Save Change: 保存对数据库的修改
- First: 定位到表中第一条记录
- Prior: 定位到当前记录的前一条记录
- Next: 定位到当前记录的下一条记录
- Last: 定位到最后一条记录

2.6.1 浏览数据

浏览数据可以使用 Data Manipulation 画板的 3 个工具按钮配合工具栏中的【First】、【Prior】、【Next】、【Last】按钮来采取不同显示风格进行浏览。

2.6.2 编辑数据

编辑数据有以下两种方法。

- (1) 在浏览状态下,选中所要编辑的字段进行编辑操作。
- (2) 在对象窗口中右击要操作的表,从快捷菜单选择【Edit Data】命令,再从中选取不同风格进行编辑。

2.6.3 排序和过滤

对于数据的排序和过滤,可通过菜单栏中【Rows】选项选择【Filter】和【Sort】命令来完成,如图 2.49 所示。

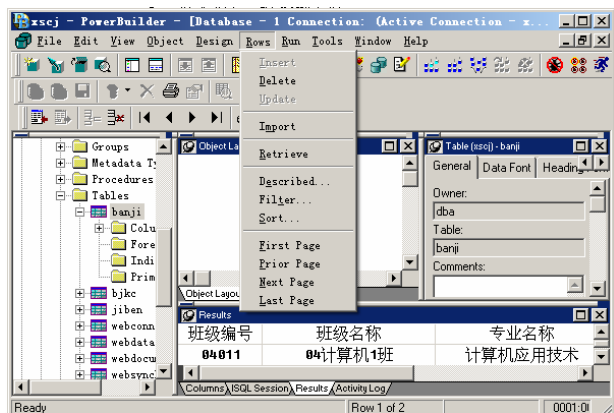


图 2.49 排序与过滤

1. 排序

操作步骤如下。

(1) 选择【Rows】|【Sort】菜单命令。

(2) 在弹出的对话框(如图 2.50 所示)中,按照提示将作为排序标准的列拖放到右边的栏中,并通过对【Ascending】复选框的设定来确定排序是以升序还是降序,如图 2.51 所示。

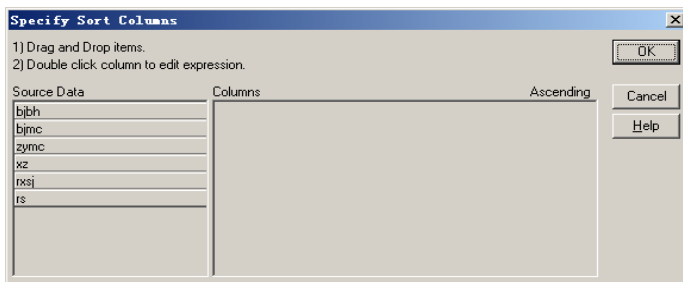


图 2.50 排序

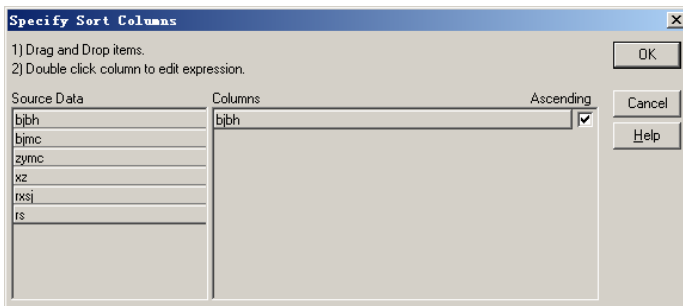


图 2.51 选择排序列

也可以根据具体情况双击右边栏中的列来进行表达式的编辑,如图 2.52 所示。

(3) 设定完毕后,按【OK】按钮完成以上步骤。

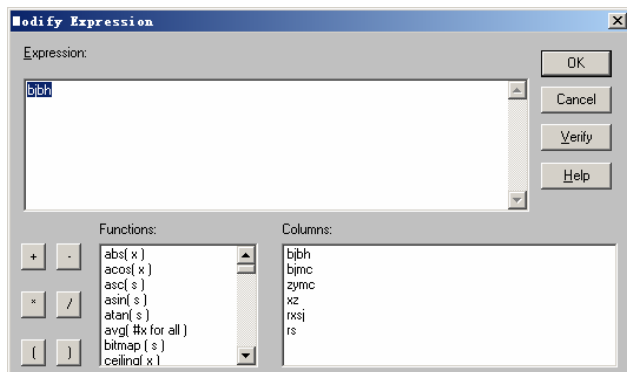


图 2.52 编辑表达式

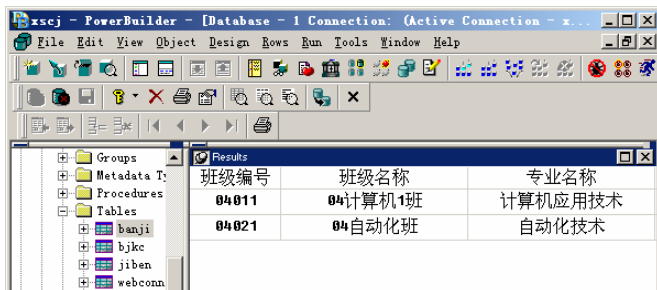
2. 过滤

操作步骤如下。

以实例进行讲解。表原始数据如图 2.53 所示。

(1) 选择【Rows】|【Filter】命令。

在弹出的对话框(如图 2.54 所示)中, 进行过滤条件表达式的编辑。要求过滤出班级编号为“04021”的班级记录。在对话框的右边选择以列“bjbh”进行过滤, 在上方的表达式编辑框中输入表达式 `bjbh="04021"`。需要注意的是, 如果写成 `bjbh=04021` 就会出错, 因为列“bjbh”的属性是“char”型的, 所以“=”号后的数据要与之类型匹配, 为“char”型。



班级编号	班级名称	专业名称
04011	04计算机1班	计算机应用技术
04021	04自动化班	自动化技术

图 2.53 表原始数据

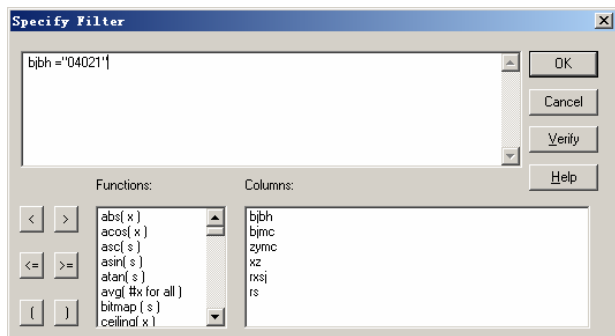
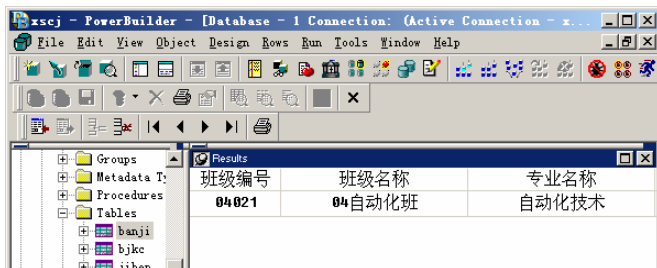


图 2.54 过滤

(2) 单击【OK】按钮完成以上操作, 结果如图 2.55 所示。



班级编号	班级名称	专业名称
04021	04自动化班	自动化技术

图 2.55 显示结果

2.7 视 图

视图是数据应用系统中经常使用的一种技术, 通过视图可以进一步简化应用程序的编

程工作。视图表面看来是一个数据库表，但实际上并不是一个真正的表。它并没有自己的数据，在数据库中也不存在它的物理结构，只存在它的定义。它就像是一个虚拟的表，它的数据是动态的，来自于其他的一个表或多个表。引用视图时，数据库会对相应的表进行查询，把符合要求的数据也就是查询结果提取出来，放入到视图中。所以在视图中所看到的只是我们向用户提供的有访问权限的一部分数据或字段，真正的表的结构或内容，用户是看不到的。这样既方便了用户，让他们只看到感兴趣的内容，也保证了数据库中表的安全性，做到信息的保密。

2.7.1 创建视图

以实例进行讲解。在“jiben”表中有很多学生的信息，我们要创建男生名单的视图。操作步骤如下。

(1) 打开数据库画板，连接要创建视图的数据库“xscj”并展开，在【Views】菜单上右击，在弹出的快捷菜单上选择【New View】菜单项，如图 2.56 所示。

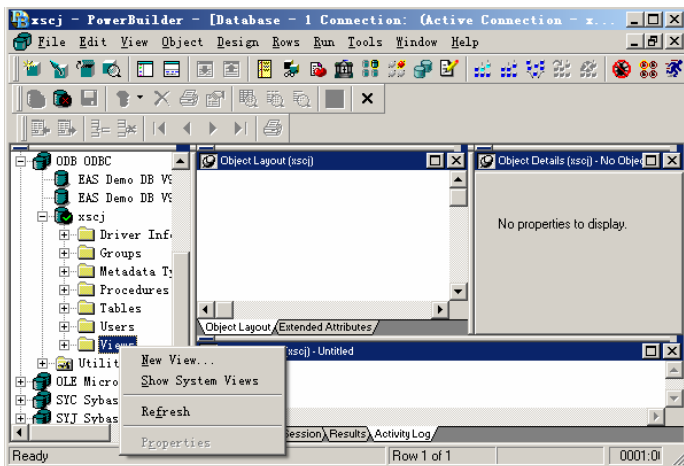


图 2.56 创建视图

接着，弹出选择视图源表对话框，如图 2.57 所示。在对话框中选择“jiben”表。单击【Open】按钮打开视图画板工作区，如图 2.58 所示。

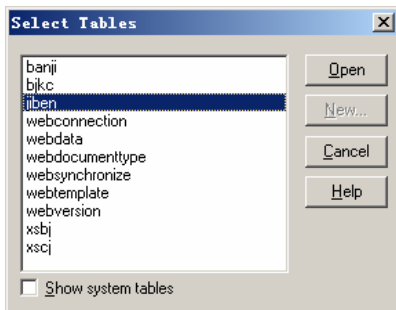


图 2.57 选择表

(2) 选择所需要的列并设定条件，如图 2.58 所示，首先选择要显示的列，这里选择全

部列。然后在下面设定查询的条件，单击标签【where】，在操作区设定条件。

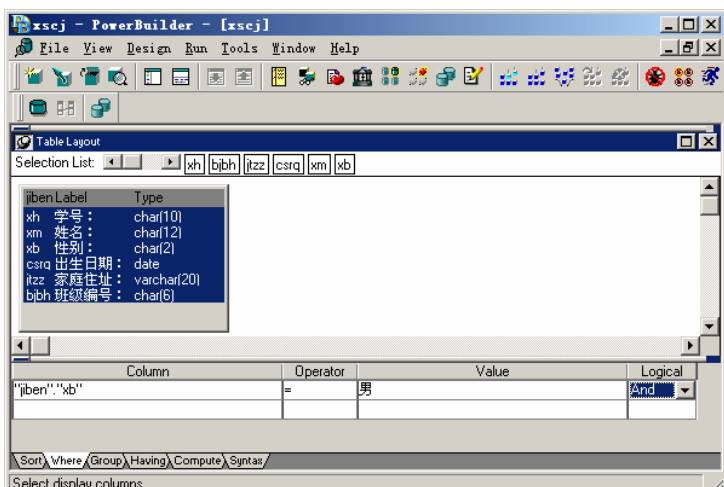


图 2.58 选择列设定查询条件

(3) 条件设定完毕后，单击工具栏上的【Return】按钮，弹出保存对话框如图 2.59 所示，输入视图名“男生名单”，然后单击【Create】按钮，创建完毕。



图 2.59 设置视图名称

在建立视图时也可以选择多个表构成视图，如图 2.60 所示。在 3 个表中选择所要显示的数据列，以构成学生成绩视图。单击工具栏上的【Return】按钮，弹出保存对话框，输入视图名为“学生成绩”，如图 2.61 所示。单击【Create】按钮，创建完成。

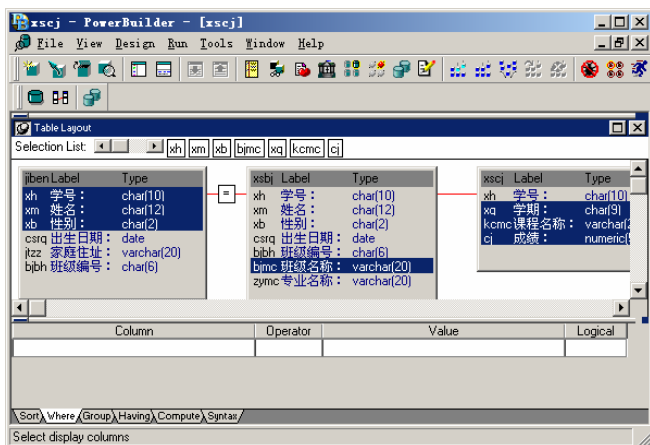


图 2.60 多个表构成视图



图 2.61 保存视图

2.7.2 删除视图

视图一旦建立好，就不能修改。如果对所建立的视图不满意，唯一能做的就是将已建立的视图删除，然后重新建立新的视图。

删除视图的方法非常简单。在数据库画板工作区，选择要删除的视图右击，在弹出的快捷菜单上选择【Drop View】命令，单击即可，如图 2.62 所示。

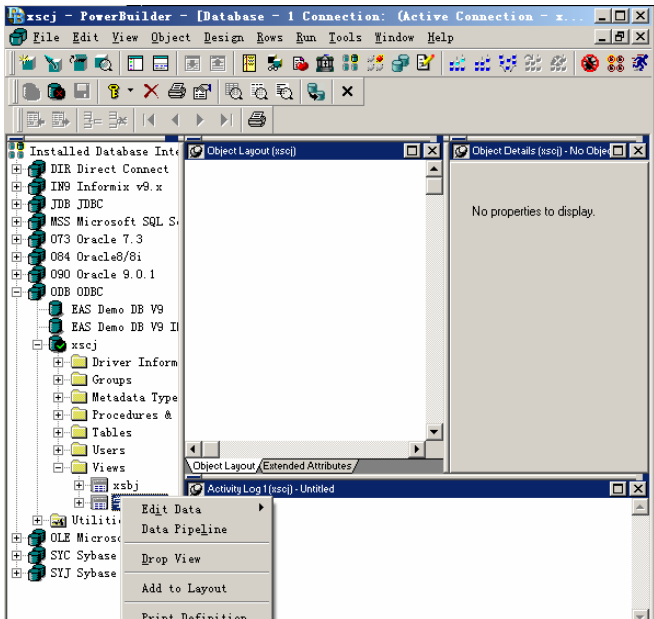


图 2.62 删除视图

2.8 数据库安全性的管理

数据库的安全性对于数据库来说至关重要，数据库当中很多数据是不可以所有人看到或者进行修改的，必须在设计时人为地进行数据库安全的设置与管理，这样才能确保数据库数据的保密性与完整性不受到影响。那么在设计时主要是通过对不同用户进行不同的访问数据库对象的权限来赋予不同用户在操作数据库时执行某种操作或访问某个对象的不同权力。以确保数据库系统的安全。

2.8.1 用户和用户组

1. 用户

1) 创建用户

创建新用户有以下两种方法。

(1) 如图 2.63 所示, 在对象操作区中展开数据库对象“xscj”, 右击【User】选项并选择【New User】菜单。弹出如图 2.64 所示对话框, 在【User】中输入用户名; 在【New Password】和【Confirm New Password】中输入用户密码, 然后单击工具栏上的【Save】按钮进行保存即可。

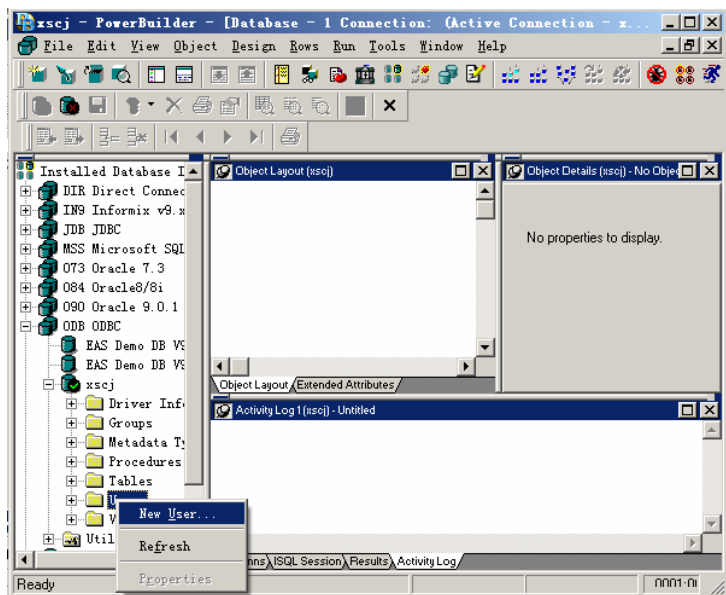


图 2.63 创建用户

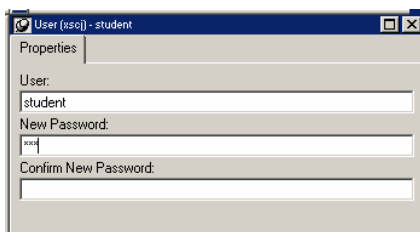


图 2.64 输入用户名及密码

(2) 如图 2.65 所示, 在对象操作区中选中数据库对象“xscj”, 从数据库画板的 Painter 的下拉列表框中选择【Create new user】选项。其他操作与第一种方法相同。

2) 修改用户密码

如图 2.66 中所示, 在对象操作区中展开数据库对象“xscj”, 展开【User】选项, 在用户【student】选项上右击并选择【Properties】选项。弹出对话框如图 2.66 所示, 在 New Password 和 Confirm New Password 中输入新的用户密码, 然后单击工具栏上的【Save】按钮

钮进行保存即可。

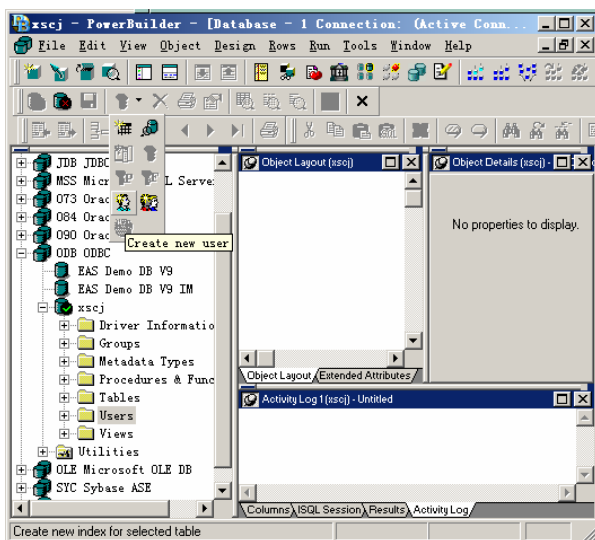


图 2.65 在工具栏上创建新用户

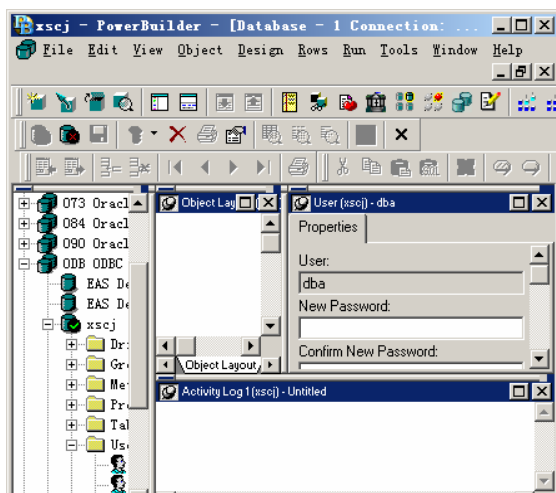


图 2.66 修改用户和密码

3) 删除用户

在图 2.67 中选择菜单【Users】，展开后在要删除的用户如【student】选项上右击，在快捷菜单上选择【Delete User】选项即可。

2. 用户组

如果一些用户对于同一个数据库有着相同的操作权限，按照前面所讲的方法，一一给他们赋予权限，进行多次重复操作非常麻烦。如同在 Word 中操作一样，为了方便地移动一组需要有着协调位置的图形，习惯上把它们先进行组合，然后再进行移动操作。这在 PowerBuilder 中也一样，为了更快捷、方便地为多个用户进行授权，也可以把多个用户定义成用户组，通过向用户组授权将相应的权限授予每一个用户。

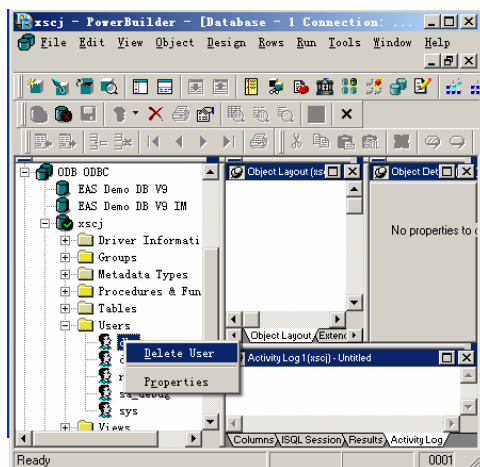


图 2.67 删除用户

1) 创建用户组

创建用户组也有两种方法。

(1) 如图 2.68 中所示，在对象操作区中展开数据库对象“xscj”，在【User】选项上右击并选择【New Group】选项。弹出对话框如图 2.69 所示，在 Group 中输入用户名；在 Members 中选择用户，然后单击工具栏上的【Save】按钮进行保存即可。

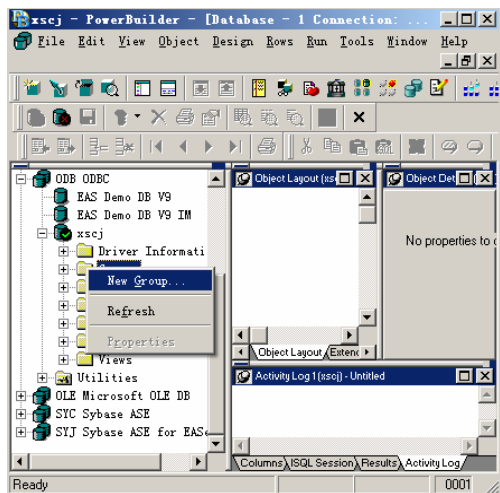


图 2.68 在对象区创建用户组

(2) 如图 2.70 所示，在对象操作区中选中数据库对象“xscj”，从数据库画板的 Painter 的下拉列表框中选择【Create new group】按钮。其他操作与第一种方法相同。

2) 修改用户组

如图 2.71 所示，在对象操作区中展开数据库对象“xscj”，展开 Groups 选项在用户组【student】上右击并选择【Properties】菜单。弹出对话框如图 2.69 所示，在 Members 中进行修改，然后单击工具栏上的【Save】按钮进行保存即可。

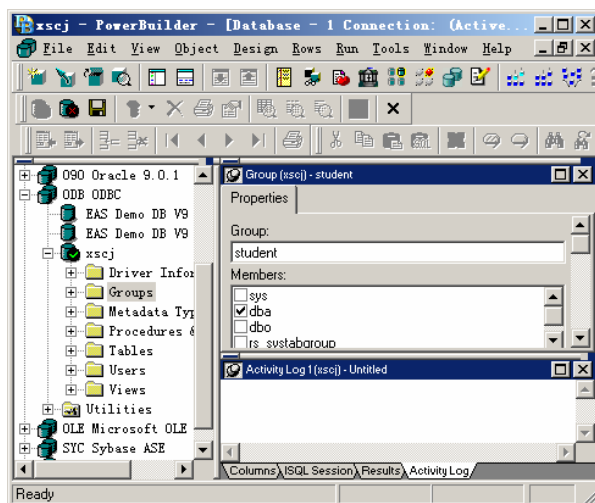


图 2.69 创建用户组

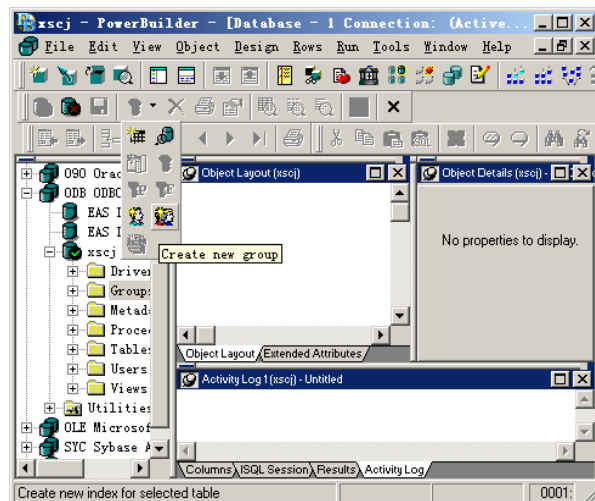


图 2.70 用工具栏按钮创建用户组

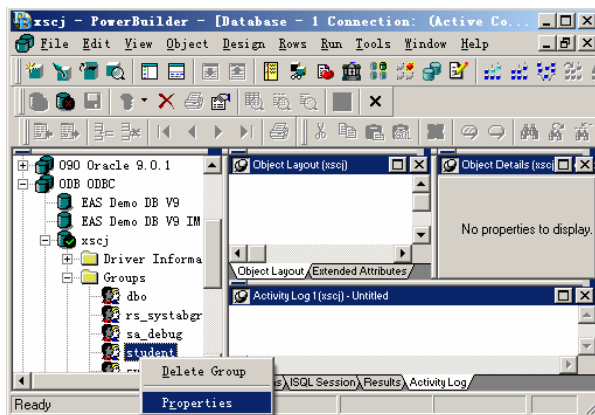


图 2.71 修改用户组

3) 删除用户组

在对象工作区中选择菜单【Groups】，展开后在要删除的用户如菜单【student】上右击，在弹出的快捷菜单上选择【Delete Group】命令即可，如图 2.72 所示。

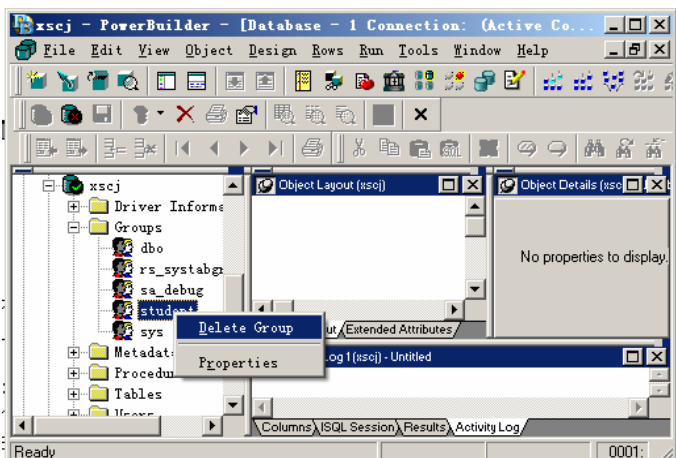


图 2.72 删除用户组

2.8.2 定义表的访问权限

创建好用户或用户组后，就可以根据不同用户或用户组的需要定义每个或每组用户对每个数据库表的访问权限。

在对象工作区中展开数据库，选择菜单【Tables】并在其上右击，在弹出的快捷菜单中选择【Table Security】命令，如图 2.73 所示。弹出表授权窗口，如图 2.74 所示。

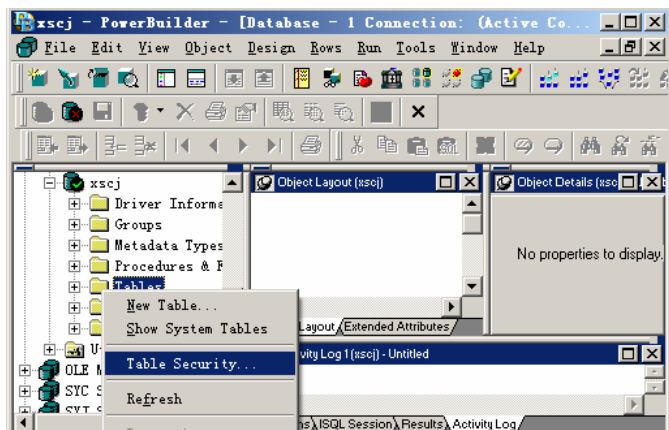


图 2.73 数据表权限设置

在【Groups/Users】列表中选择用户或用户组，在【Tables】列表中选择数据库表，在【Privileges】列表中选择该用户或用户组对该表的操作权限。如果只对该表中的某个或某些列指定权限，只需单击【Update】按钮。在图 2.75 中，从【Select Columns】列表中选择具有修改权限的列，单击【OK】按钮即返回。

完成所有设置后，在如图 2.74 所示的窗口中单击【Apply】按钮即可。

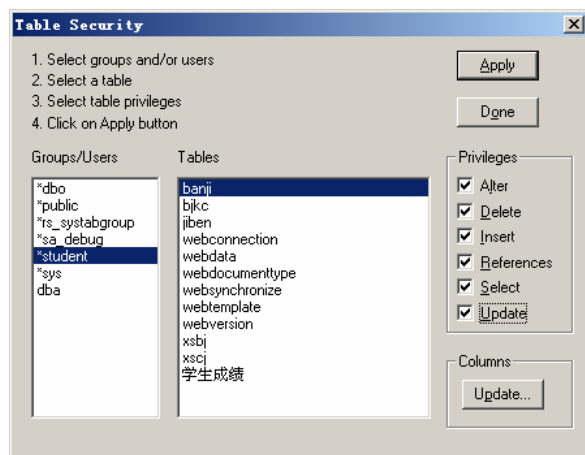


图 2.74 授权窗口

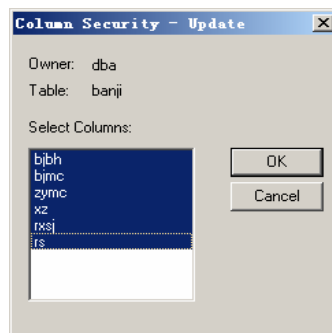


图 2.75 列授权

2.9 小 结

在本章，我们详细介绍了数据库基础知识及数据库的设计步骤和方法。

PowerBuilder 9.0 提供了丰富的数据库接口来支持各种数据库管理系统，其中既包括通用接口如 ODBC，也包括专用接口如 SQL Server 等。由于 ODBC 接口具有通用性，所以在本章中，我们主要介绍了创建 ODBC 数据源的方法及基于 ODBC 接口的数据库描述文件的如何创建。

本章中我们还详细介绍了数据库、表、主键、外键、索引、视图的作用与创建方法。

PowerBuilder 9.0 的数据库画板是功能强大的数据库操作平台，我们详细介绍了在此平台下的数据的操作方式。

掌握数据库的安全性的管理方法是确保数据库系统的安全所必需的。

通过对本章的学习，相信大家对 PowerBuilder 9.0 和数据库的相关操作有了一定的了解。

2.10 实 训

实训目的

通过本节实训，主要学习以下操作。

- (1) 数据库的创建、删除操作。
- (2) 掌握表的创建、修改、删除和属性设置操作。
- (3) 掌握主键、外键、索引的建立、修改、删除和属性设置操作。
- (4) 掌握视图的创建、删除和属性设置操作。
- (5) 掌握用户、用户组的创建，修改，删除操作及权限设置操作。

实训内容

建立一个小型数据库，实现班级学生学习成绩的管理。

实训步骤

创建学生成绩管理系统的 ASA 数据库 xscj，存储路径为“d:\xscj\data\xscj.db”。

根据提供的学生成绩管理系统的 4 个基本表格的结构见表 2-5~表 2-8。

(1) 班级(banji)。

表 2-5 班级表列属性

列名	类型	宽度	null	解释
bjbh	char	6	no	班级编号
bjmc	varchar	20	no	班级名称
zymc	varchar	20	no	专业名称
xz	numeric	1.0	no	学制
rxsj	date		no	入学时间
rs	numeric	3.0	yes	人数

(2) 学生基本信息(jiben)。

表 2-6 学生基本信息表列属性

列名	类型	宽度	null	解释
xh	char	10	no	学号
xm	char	12	no	姓名
xb	char	2	no	性别
csrq	date		yes	出生日期
jtzz	varchar	20	yes	家庭住址
bjbh	char	6	yes	班级编号

(3) 班级课程(bjkc)。

表 2-7 班级课程表列属性

列名	类型	宽度	null	解释
Bjbh	char	6	no	班级编号
Xq	char	9	no	学期
Kcmc	varchar	20	no	课程名称
Xs	numeric	3.0	no	学时
jsxm	char	12	yes	教师姓名

(4) 学生成绩(xscj)。

表 2-8 学生成绩表列属性

列名	类型	宽度	null	解释
xh	char	10	no	学号
xq	char	9	no	学期
kcmc	varchar	20	no	课程名称
cj	numeric	5.1	yes	成绩

创建数据库的 4 个表并为 4 个基本表创建主键、外键和索引，具体要求见表 2-9。

表 2-9 具体要求表

Table	Primary key	Foreign key Name	Foreign key Columns	Index
banji	bjbh			
bjkc	bjbh	bjbh	bjbh	
	xq			
	kcmc			
jiben	xh	bjbh	bjbh	
xscj	xh	xh	xh	
	xq			
	kcmc			

对每个基本表进行数据录入，录入 5~10 个记录。

创建学生班级的视图(名称为“xsbj”), 显示结果包括 xh(学号), xm(姓名), xb(性别), csrq(出生日期), bjbh(班级编号), bjmc(班级名称), zymc(专业名称)。条件: banji.bjbh = jiben.bjbh。

创建用户: dba, dbo, rs_systabgroup, sa_debug, sys 并为其授权。

创建用户组: dbo, rs_systabgroup, sa_debug, sys 并为其授权。

2.11 习 题

1. 简述 PowerBuilder 可以使用什么样的数据库?
2. 数据源的作用是什么? 如何创建数据源?
3. 如何创建 ASA 数据库?
4. 表的主要作用是什么? 主键和外键及索引的作用分别是什么?
5. 简述视图的作用及特点。
6. 如何确保数据库系统的安全性?