



使用 JavaServer Faces 在 Web 页面上显示动态信息

目录

使用 JavaServer Faces 在 Web 页面上

显示动态信息 1

使用 JavaServer Faces 在 Web 页面上显示动态信息 . 1

模块 1: 创建具有数据连接的 Web 页面 4

练习 1.1: 导入必需的资源 and 设置目标服务器 5

练习 1.2: 连接至数据库和在 Web 页面上显示数据 . 6

课程 1.3: 测试 Web 站点 13

课程 1.4: 创建新记录以显示和更新数据库 14

课程 1.5: 对提交按钮进行编程 18

课程 1.6: 创建更新页面 20

模块 1: 总结 24

模块 2: 添加高级功能 25

课程 2.1: 格式化数据表 29

练习 2.2: 使用文件上传组件 35

课程 2.3: 使用导航规则 37

课程 2.4: 使用自动键生成 39

模块 2: 总结 41

使用 JavaServer Faces 在 Web 页面上显示动态信息 . 42

使用 JavaServer Faces 在 Web 页面上显示动态信息

本教程将介绍如何设计一个动态 Web 站点，其功能相当于报纸的分类信息广告部分。该 Web 站点是一个完整的 J2EE Web 应用程序，它使用了一些 JavaServer Faces (JSF) 组件和“服务数据对象” (SDO)。此应用程序使用 JSF 技术来创建动态 Web 页面，这些 Web 页面链接至一个用来存储分类信息广告数据的数据库。通过将 JSF 组件拖动到 Web 页面上，可以创建称为 CRUD 应用程序的 JSF Web 应用程序。之所以将它称为 CRUD 应用程序，是因为它可以在数据库中创建记录，还可以读取、更新和删除数据库中的记录。Web 站点的用户可以使用这些功能来管理数据库中的分类信息广告。

学习目标

本教程分为两个模块，每个模块都有它自己的学习目标。您可以选择完成其中一个模块，也可以完成这两个模块。在每个模块中，必须完成其中的练习才能使 Web 站点正常工作。

模块 1: 创建具有数据连接的 Web 页面

此模块将介绍如何与数据库建立连接并在 Web 页面上显示来自该数据库的信息。在本模块中，您将完成下列任务：

- 将 Web 页面连接至数据库。
- 创建一些页面，可以通过它们显示、创建、编辑和删除数据库中的记录。
- 将数据从一个页面发送到另一个页面。

模块 2: 添加高级功能

此模块将介绍一些功能更强大的方法来使用数据库中的数据。在本模块中，您将完成下列任务：

- 格式化 Web 页面上的数据库记录
- 添加一个组件，它允许您将文件从 Web 页面上载到数据库中
- 从一个页面自动浏览至另一个页面
- 自动执行某些数据库管理任务

当一切准备工作就绪之后，就可以开始学习“模块 1: 创建具有数据连接的 Web 页面”。

所需时间

所需总时间: 3 小时 30 分钟

模块 1: 2 小时

模块 2: 1 小时 30 分钟

使用 JavaServer Faces 在 Web 页面上显示动态信息

本教程将介绍如何设计一个动态 Web 站点，其功能相当于报纸的分类信息广告部分。该 Web 站点是一个完整的 J2EE Web 应用程序，它使用了一些 JavaServer Faces (JSF) 组件和“服务数据对象” (SDO)。此应用程序使用 JSF 技术来创建动态 Web 页面，这些 Web 页面链接至一个用来存储分类信息广告数据的数据库。通过将 JSF 组件拖动到 Web 页面上，可以创建称为 CRUD 应用程序的 JSF Web 应用程序。之所以将它称为

CRUD 应用程序，是因为它可以在数据库中创建记录，还可以读取、更新和删除数据库中的记录。Web 站点的用户可以使用这些功能来管理数据库中的分类信息广告。

此教程可能需要一些可选可安装组件。要确保您已安装了适当的可选组件，请参阅“系统要求”列表。

完成本教程之后，使用此站点的用户不仅能够查看待售商品，而且可以添加新商品、更改有关商品的详细信息（例如，价格或者对商品的描述）或者搜索特定商品。虽然本教程中描述的是一个简单设计的站点，但是它使用的原理和技术同样适用于更复杂的大型 Web 站点。

学习目标

本教程分为两个模块，每个模块都有它自己的学习目标。您可以选择完成其中一个模块，也可以完成这两个模块。在每个模块中，必须完成其中的练习才能使 Web 站点正常工作。

模块 1：创建具有数据连接的 Web 页面

此模块将介绍如何与数据库建立连接并在 Web 页面上显示来自该数据库的信息。在本模块中，您将完成下列任务：

- 将 Web 页面连接至数据库。
- 创建一些页面，可以通过它们显示、创建、编辑和删除数据库中的记录。
- 将数据从一个页面发送到另一个页面。

模块 2：添加高级功能

此模块将介绍一些功能更强大的方法来使用数据库中的数据。在本模块中，您将完成下列任务：

- 格式化 Web 页面上的数据库记录
- 添加一个组件，它允许您将文件从 Web 页面上载到数据库中
- 从一个页面自动浏览至另一个页面
- 自动执行某些数据库管理任务

当一切准备工作就绪之后，就可以开始学习“模块 1：创建具有数据连接的 Web 页面”。

所需时间

所需总时间：3 小时 30 分钟

模块 1：2 小时

模块 2：1 小时 30 分钟

熟练程度

中级

用户

Web 应用程序开发者和 Web 用户界面设计人员

系统要求

要完成本教程，必须首先安装并配置运行时服务器。已经使用下列服务器测试了本教程：

- WebSphere® Application Server 6.1

2 使用 JavaServer Faces 在 Web 页面上显示动态信息

有关在 WebSphere Application Server 之外的服务器上部署 SDO 的信息，请参阅帮助主题：在非 WAS 服务器上部署 SDO。

本教程可能需要在安装产品期间尚未启动的一些能力。如果找不到本教程中所描述的用户界面选项，则应确保启用了适当的功能（窗口 → 首选项 → 能力）。为了完成本教程，您将需要具备下列能力：

- Java™ 开发者
- Web 开发者（典型）
- XML 开发者
- 数据库开发者
- 企业 Java 开发者
- 数据

要使用此教程，必须安装并配置了应用程序服务器。要验证是否提供了服务器运行时环境，请单击窗口 → 首选项，展开服务器，然后单击已安装运行时。可以使用此窗格来添加、除去或编辑已安装的服务器运行时定义。还可以下载并安装对新服务器的支持。

先决条件

要完成本教程，您应该熟悉下列内容：

- 基本的 Web 设计概念，例如，Web 站点、Web 页面、浏览器和服务器。
- 如何创建简单的静态 Web 页面。
- Web 页面的元素，例如，表、超链接、表单和图像。
- 数据库术语，例如，表、记录、列和字段。

了解下列内容也会有所帮助：

- 如何使用工作台的透视图和视图。
- 如何编辑 Web 页面的 HTML 代码。

期望的结果

一旦完成了该 Web 应用程序，其主页将类似于下图。通过站点模板对 Web 页面添加的 Web 站点导航将链接至一些页面，这些页面允许您创建新的广告或者查看所有广告。通过对下面所显示的表中的行执行操作将显示一个更新页面，可以在此页面中更改特定广告的相关信息。

Home

View all listings »

Post a listing

View all listings

Title	Details	Category
WV Passat wagon 	Description: 2000 good condition Price: \$1,000.00 Phone: 416-512-8665	Automotive
Magic Flute 	Description: Imagine it. Price: \$101.00 Phone: 314-159-265	Entertainment
Video Camera 	Description: Great camera, suitable for amateur or professional use. Price: \$900.00 Phone: 534-333-454	Entertainment
VCR 	Description: Old and worn but still working ok, hence low price. Price: \$40.00 Phone: 322-654-044	Entertainment
Flat Screen TV 	Description: Very clear picture, sound excellent too. Reluctant sale. Price: \$700.00 Phone: 456-456-944	Entertainment
1 2 3 4 5		

相关信息



查看本教程的 PDF 版本

模块 1: 创建具有数据连接的 Web 页面

在此模块中，您将学习如何创建一些 Web 页面，可以通过这些 Web 页面来显示、编辑、删除和创建数据库中的信息。您将学习 JavaServer Faces (JSF) 技术并通过它使复杂的编程任务简单化。

学习目标

此模块将介绍如何与数据库建立连接并在 Web 页面上显示来自该数据库的信息。在本模块中，您将完成下列任务：

- 将 Web 页面连接至数据库。
- 创建一些页面，可以通过它们显示、创建、编辑和删除数据库中的记录。

4 使用 JavaServer Faces 在 Web 页面上显示动态信息

- 将数据从一个页面发送到另一个页面。

所需时间

完成此模块大约需要 2 个小时。如果您决定在学习本教程的同时研究动态 Web 站点的其他方面，则可能要花更长时间才能完成本教程。

练习 1.1: 导入必需的资源 and 设置目标服务器

在开始之前，需要导入本教程需要的资源：一系列 Web 页面和一个样本 Derby 数据库。

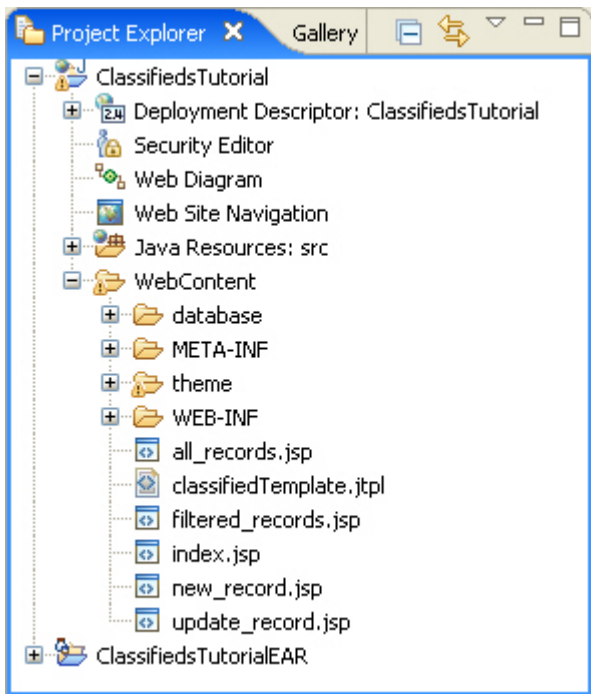
本教程的主要目的是介绍如何创建一个 Web 应用程序，从而使用户能够访问和处理数据库中的数据。本教程并不会详细描述 Web 站点外观的设计。因此，已经为您准备好了 Web 站点设计。

要完成本教程，需要访问一组现有的 JavaServer Faces JSP 文件和一个样本 Derby 数据库。这些基础 Web 页面和样本数据库都包含在一个 ZIP 文件中。本课程将指导您完成导入该 ZIP 文件所需要执行的步骤，以便可以使用这些 Web 页面和数据库。您还将设置动态 Web 项目的目标服务器。

将用于本教程的 Web 页面和数据库包含在一个 ZIP 文件中。要导入该 ZIP 文件的内容：

导入样本项目文件

1. 导入项目。切换至 Web 透视图（窗口 → 打开透视图 → **Web**）。
2. 在 Web 透视图的“项目资源管理器”视图中，确保 ClassifiedsTutorial 项目类似于下图：



设置动态 Web 项目的目标服务器

设置 Web 项目的目标服务器使您能够测试将在本教程中创建的资源。

要设置目标服务器：

1. 在 Web 透视图的“项目资源管理器”视图中，右键单击 **ClassifiedsTutorial** 并选择属性。

2. 在属性列表中，单击**服务器**。
3. 在“缺省服务器”列表中，选择想要使用的缺省服务器。单击**应用**。
4. 在属性列表中，单击**目标运行时**。
5. 在“运行时”列表中，单击与您所选择的服务器相对应的运行时。单击**确定**。

如果目标运行时列表中未列示您想使用的服务器，则关闭工作台并安装想要使用的服务器。一旦安装了该服务器，则遵循指示信息来设置目标服务器。

注：如果未看到“缺省服务器”列表中列示任何服务器，但是您已经安装了服务器运行时，则可能需要配置服务器。要配置服务器，可以执行下列操作：

1. 右键单击 **all_records.jsp** 文件，然后单击**运行方式** → **在服务器上运行**。
2. 选择**手工定义新的服务器**。
3. 选择已安装的服务器。
4. 遵循向导中的指示来配置服务器。当您首次在服务器上运行时，可能会接收到错误。要修正该错误，按以上所述设置目标服务器，在“服务器”视图中重新启动服务器，并在浏览器中重新装入 Web 页面。

如果重新执行前面所描述的步骤来设置目标服务器，则会发现缺省服务器就是您刚才所配置的服务器。如果已经安装但是未配置服务器，则该服务器将不会显示在服务器列表中，您可以从该列表中选择缺省目标服务器。

有关在 WebSphere Application Server 之外的服务器上部署 SDO 的信息，请参阅帮助主题：在非 WAS 服务器上部署 SDO。

课程要点

现在，就已经导入了 ClassifiedsTutorial 动态 Web 项目并且设置了目标服务器。

可以浏览教程 Web 项目中的文件。要打开某个文件，在“项目资源管理器”视图中双击该文件。要查看用类似站点图表示的页面之间的相关性，双击“项目资源管理器”中的 Web 站点导航。

您在此样本中的主要工作将涉及到下列文件：

all_records.jsp

这是站点的主页。它将显示数据库中的每一条分类信息广告。

new_record.jsp

此页面将创建新的分类信息广告。

update_record.jsp

此页面将更改有关数据库中广告的详细信息或者删除广告。

classifiedTemplate.jtpl

这是站点页面的模板。它包括诸如出现在每个页面上的表和“欢迎使用分类信息广告”灰色条幅之类的元素。此页面在灰色条幅下方还有两个导航标签，分别引导至主页和新的分类信息广告页面。

现在，您可以开始学习“练习 1.2：使用关系记录列表和数据表组件”。

练习 1.2：连接至数据库和在 Web 页面上显示数据

在此课程中，将学习如何连接至数据库以及在 Web 页面上显示数据记录。还将学习如何将关系记录列表添加至 Web 页面。

本教程中的 Web 站点将动态 Web 页面与 JavaServer Faces 组件配合使用来访问数据库和在 Web 页面上显示信息。本教程使用关系记录和关系记录列表来表示数据库中的数据，以便可以采用数据表或普通 HTML 表的形式将这些数据显示在页面上。这些组件将使用 Java 访问 bean。

更多地了解数据访问 bean:

数据访问 bean 是企业 bean 的 Java bean 表示。它们通常用于那些采用 JavaServer Pages (JSP) 文件、servlet 或其他企业 bean 建立接口的企业 bean 的客户机程序中。访问 bean 可以使您不必考虑管理企业 bean 生命周期的复杂性。这意味着可以像编写 Java bean 一样很容易地编写企业 bean，可以简化企业 bean 客户机程序并且缩短总的开发时间。

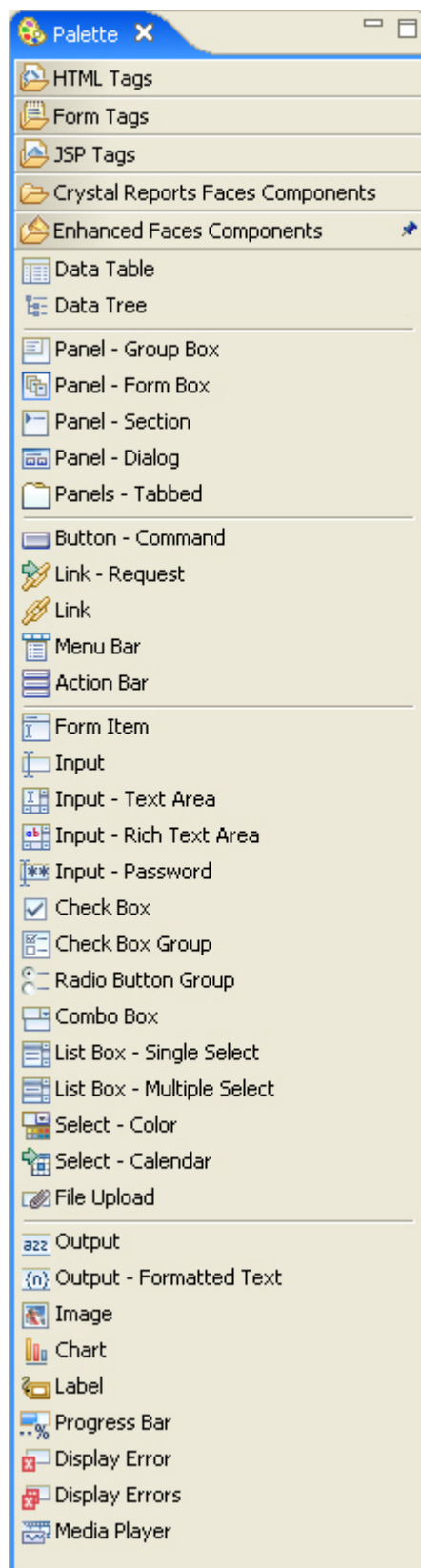
了解有关 JavaServer Faces 和 Faces 组件的更多信息:

JavaServer Faces 是一项技术，可帮助您为在服务器上运行的动态 Web 应用程序构建用户界面。JavaServer Faces 框架管理各服务器请求中的用户界面状态，并提供了一个用于开发由客户机激活的服务器端事件的简单模型。JavaServer Faces 基于“模型 - 视图 - 控制器”（MVC）框架。对于 JavaServer Faces，这意味着控制器是 servlet，模型由 JavaBeans™ 表示，而视图由带很少或不带应用程序代码的 JavaServer Faces 组件组成。此模型的目标是将内容与表示形式分开。

一些诸如“Faces 组件”的工具帮助您在 Web 应用程序中使用此技术。“Faces 组件”包括一个 JavaServer Page (JSP) 定制标记库，用于在页面中表示 JavaServer Faces 界面。此向导帮助您创建可使用 Faces 组件的 JSP 文件。Faces 组件允许您通过将组件从“选用板”视图的 Faces 抽屉拖放到正在创建的 Web 页面上，从而创建 Web 应用程序。

例如，可以将输入文本字段拖放到页面上的表单中。然后，可以将提交按钮拖放到输入文本字段旁边。最后，可以将输入文本字段连接至数据源。这将使最终用户能够将 Web 应用程序中的数据输入到数据源中。

使用 Faces 组件构建的应用程序的另一个优点是：用户界面的呈现和底层程序逻辑无关。这意味着应用程序可以在不同平台（例如，浏览器或手持设备）上运行和访问数据。



添加关系记录列表

在本课程中，将创建一个关系记录列表来表示数据库中的所有分类信息广告。然后，将连接至数据库，并在关系记录列表中选择用来保存所需信息的表。最后，将在页面上的数据表中显示此关系记录列表。

- 关系记录只连接至数据库中的一条记录。在本例中，关系记录表示样本数据库中的单条分类信息广告。通过使用关系记录，可以创建新记录、编辑现有记录或者删除旧记录。
- 关系记录列表连接至数据库中的多条记录。在本例中，关系记录列表表示样本数据库中的两个到全部的分类信息广告。
- 数据表将关系记录列表中的数据显示在页面上。数据表只为记录列表指定一个位置，它们不会像 HTML 表那样将数据格式化成许多行和列。

更多地了解数据表：

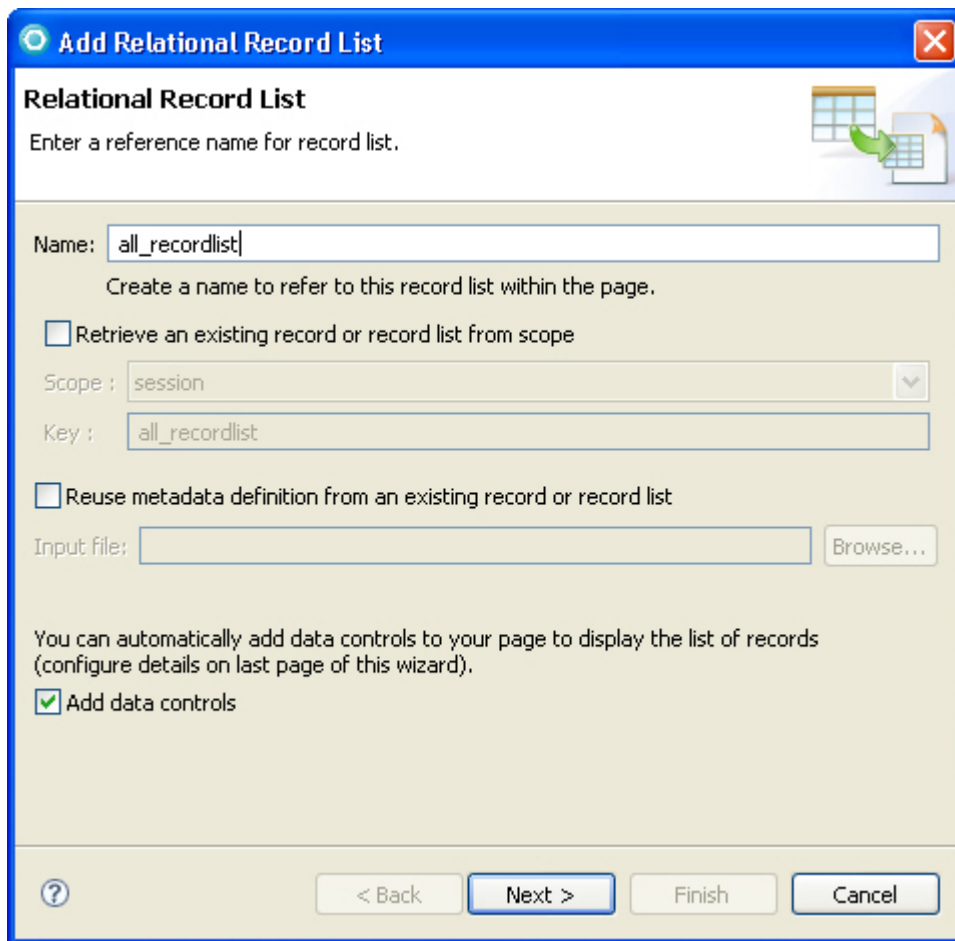
数据表是 Faces JSP 页中的一个组件，它用来保存一些数据对象（例如，关系记录列表）。虽然数据表看起来与 HTML 表一样具有一些行和列，但是它的工作方式与 HTML 表并不相同。如果想要像在表中那样定义输入和输出控件的格式，则必须使用“选用板”中的“增强型 Faces 组件”抽屉中的“面板 - 组框”。在“练习 2.2：格式化数据表”中对此进行了描述。

因为数据表是 Faces 组件而不是 HTML 组件，所以是通过“属性”视图而不是 Page Designer 视图来控制。通过使用“属性”视图，可以使用各种方法来定制数据表：

- 添加、除去和更改列的顺序
- 格式化文本和背景
- 添加页眉、页脚和页边距
- 对结果显示添加分页

要创建新的关系记录列表：

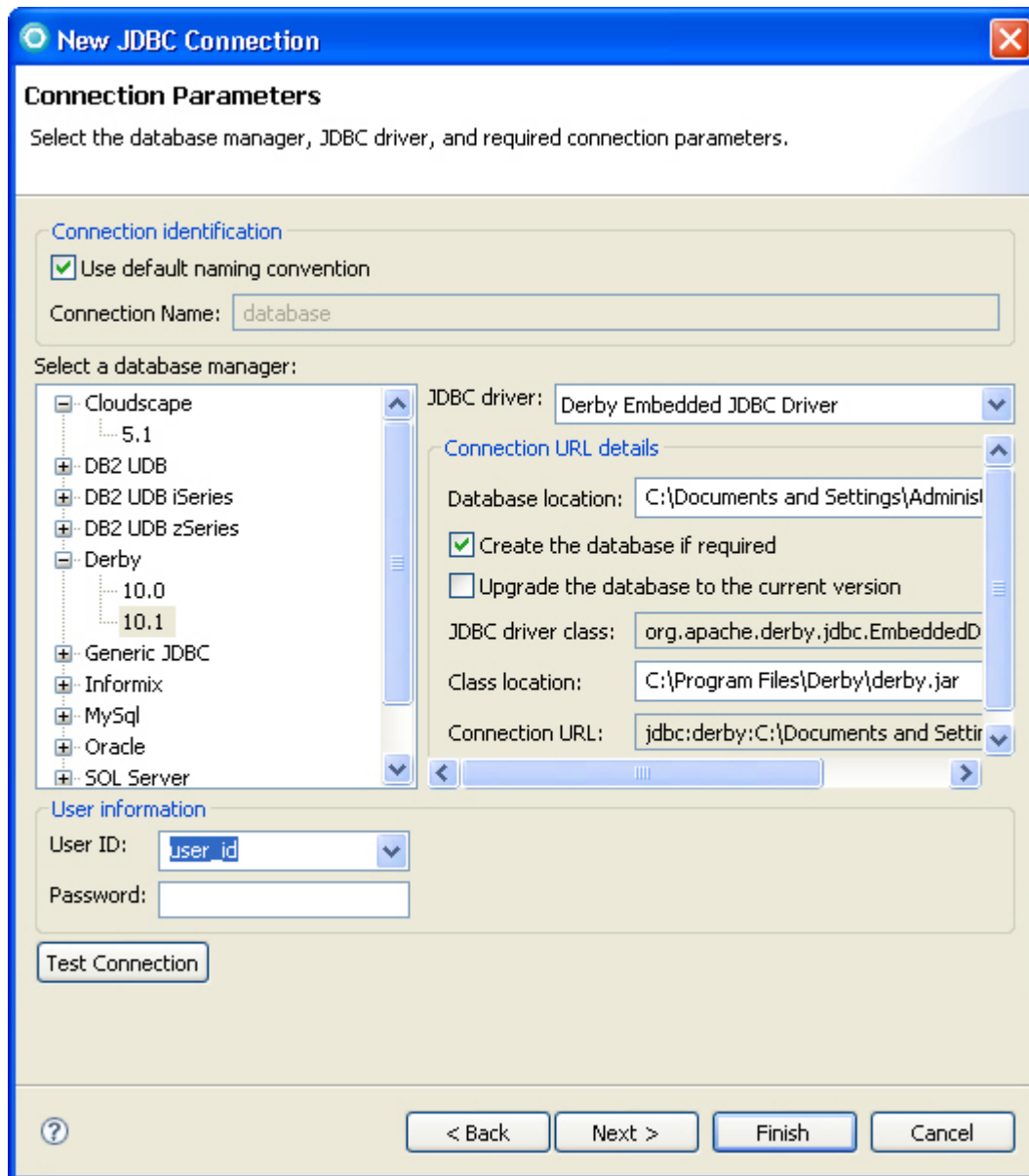
1. 在 Web 透视图的“项目资源管理器”视图中，展开 **ClassifiedsTutorial** → **WebContent**。
2. 在 WebContent 文件夹中双击 **all_records.jsp**。all_records.jsp 文件就会在编辑器中打开。
3. 删除缺省文本区域的缺省内容。
4. 在“选用板”视图中，展开**数据抽屉**。
5. 将**关系记录列表**组件从“选用板”拖放到空白内容区域上。将打开“添加关系记录列表”窗口。
6. 在“名称”字段中，输入 **all_recordlist**。关系记录列表和关系记录的名称必须符合变量名的 Java 标准命名约定。例如，这些名称不能包含任何空格。
7. 确保选择了**添加数据控件**。如果选择了**添加数据控件**，向导就会创建数据表以在页面上显示记录列表。否则，向导将只创建记录列表而不创建该数据在页面上的表示。现在，向导将创建缺省数据表，稍后您将定制该数据表。“添加关系记录列表”窗口应类似于下图：



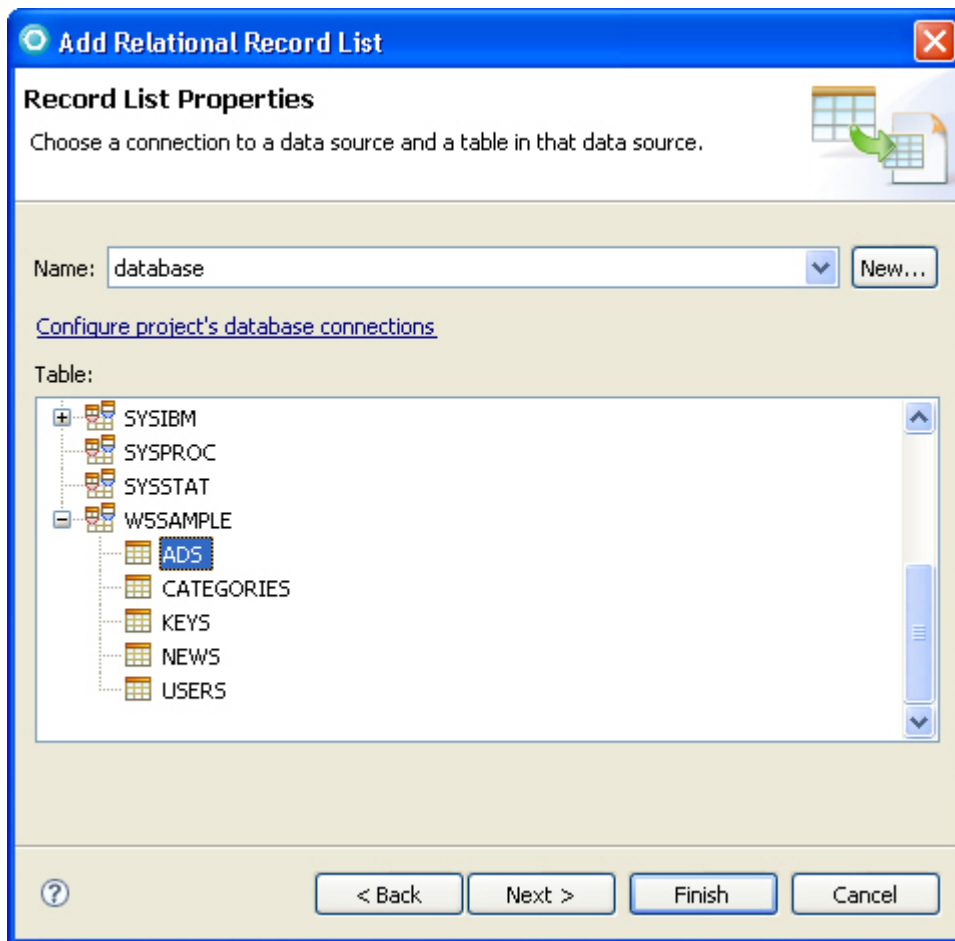
。

单击下一步。

8. 在向导下一页上的**连接名称**字段中，单击**新建**以创建新的数据库连接。将打开新建 JDBC 连接对话框。选择**创建新的连接**单选按钮，然后单击**下一步**。
9. 在**选择数据库管理器**列表中，展开 **Derby**，然后选择 **10.1**。
10. 在“**数据库位置**”字段中，单击**浏览**并选择 `<workspace_location>\ClassifiedsTutorial\WebContent\database`，其中 `<workspace_location>` 是当前工作空间的目录。单击**确定**以关闭浏览以找到文件夹窗口。
11. 在**类位置**字段中，输入用来实现 JDBC 驱动程序类的一个或多个 JAR 所在的位置，或者单击**浏览**以选择文件系统中的某个位置。选择了 JAR 文件的位置之后，单击**打开**。
12. 可能需要输入用户标识才能访问数据库。但密码不是必需的。“新建数据库连接”向导应类似于下图：



13. 在“新建 JDBC 连接”向导中，单击**完成**。您将返回到添加关系记录列表向导。
14. 单击**配置项目的数据库连接**。将打开“JDBC 连接属性”窗口。
15. 单击“运行时连接详细信息”部分旁边的**编辑**。选择**使用驱动程序管理器连接**作为运行时连接类型。单击**完成**，然后单击**确定**以返回到“添加关系记录列表”对话框。既然您已经创建了与 Derby 数据库的连接，就需要选择要表示的表或记录列表。“添加关系记录列表”向导显示数据库中的表。
16. 展开 **W5SAMPLE** 并选择 **ADS**。单击**下一步**。向导中的其余页面使您能够从记录列表中排除列并执行高级选项，例如，定义主键、添加与其他表的关系以及指定过滤和排序条件。您将在后面的练习中了解有关这些页面的更多信息。

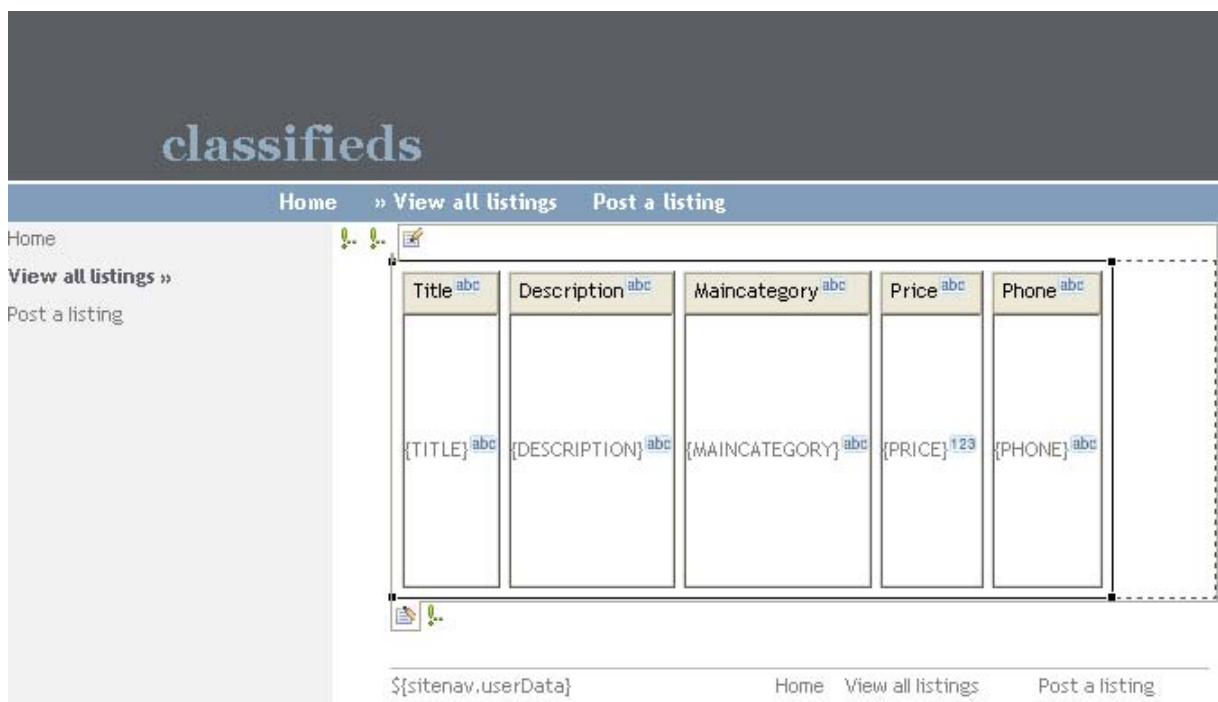


17. 在该对话框的添加关系记录页面中，您将指定想要包含在关系记录列表中的列。单击下一步。
18. 在本教程中，只需要显示每个分类信息广告的标题、描述、类别、价格和电话号码。缺省情况下，将选择所有列，单击以清除下列各列之外的所有列：
 - **MAINCATEGORY: String**
 - **TITLE: String**
 - **DESCRIPTION: String**
 - **PRICE: BigDecimal**
 - **PHONE: String**
19. 可以进一步优化 Web 页面上显示的数据表的组织形式。要将各列重新排序，选择一列并单击向上箭头和向下箭头，直到该列位于期望位置为止。

按照以下顺序排列这些列：

- **TITLE**
- **DESCRIPTION**
- **MAINCATEGORY**
- **PRICE**
- **PHONE**

单击**完成**。您的 Web 页面现在看起来将类似于下图：



20. 保存该页面。

在下一课程中，您将看到此页面在实际 Web 服务器上的外观。有许多用于格式化数据表和 JavaServer Faces 组件的选项。某些选项包含在下一个模块中。您自己也可以研究页面上的各种 JavaServer Faces 组件（例如，数据表和各个输出组件）的“属性”视图。

课程要点

您已经完成了课程 1.2。在此课程中，您学习了如何连接至数据库以及在 Web 页面上使用关系记录列表来显示数据记录。

课程 1.3: 测试 Web 站点

当您准备发布 Web 应用程序时，将需要一个主管该 Web 应用程序的服务器，以使用户能够通过因特网访问 Web 站点。但是，要测试该 Web 站点，可以使用一个测试环境。

在本课程中，将在测试环境中运行 JSP 来模拟一个页面在实际发布时将具有的外观。还将学习如何启动和停止服务器。在 Web 站点开发过程中的任何时候，您都可以在 Page Designer 中打开页面，并使用预览选项卡来查看您的设计在浏览器中的效果。但是，“预览”视图不允许您查看页面在服务器上运行时显示的动态内容（例如，数据库连接）。

提示：一旦启动了服务器，就必须停止它才能继续使用站点。

要使用此教程，必须安装并配置了应用程序服务器。要验证是否提供了服务器运行时环境，请单击窗口 → 首选项，展开服务器，然后单击已安装运行时。可以使用此窗格来添加、除去或编辑已安装的服务器运行时定义。还可以下载并安装对新服务器的支持。

要在服务器上测试 Web 站点:

1. 在“项目资源管理器”中，右键单击 **all_records.jsp** 文件，然后单击运行方式 → 在服务器上运行。将打开在服务器上运行窗口。

2. 必须使用在练习 1.1 中选择的用作目标服务器的那个服务器。在此步骤中使用另外一个服务器会产生错误。
 - 如果要在先前定义的服务器配置上测试 Web 站点:
 - a. 选择**选择现有服务器**。
 - b. 从提供的列表中选择想要使用的服务器。
 - 如果要在新的服务器配置上测试 Web 站点:
 - a. 选择**手工定义新的服务器**。
 - b. 从提供的列表中选择想要使用的服务器类型。
3. 单击**完成**。

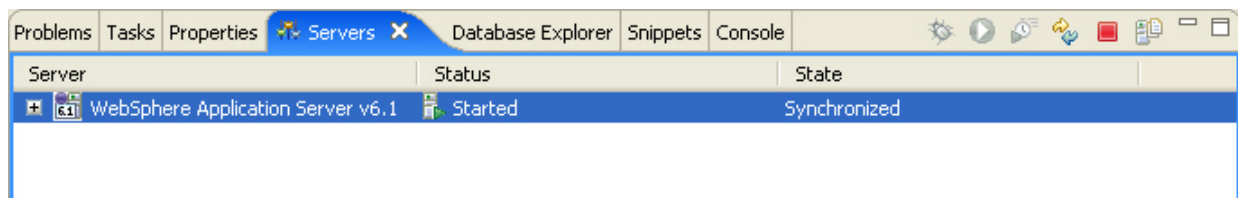
服务器工具会创建新的服务器，启动它并在工作台的内部 Web 浏览器中打开页面。这可能要花一点时间。在“控制台”视图中，当服务器工具启动服务器时可以看到一些消息。

完成此过程之后，您可以预览到一旦真正将站点发布到 Web 服务器上时该站点的外观和工作方式。单击链接、在表单中输入数据以及查看站点中动态生成的但可能未显示在“预览”视图中的任何内容。另外，如果想要在其他 Web 浏览器中查看页面，可以将 URL 从工作台 Web 浏览器复制到您选择的浏览器中。

必须停止服务器才能继续完成本教程。如果使服务器保持运行，则在继续完成本教程时将产生错误消息。服务器已连接至样本数据库，因此出现了这些错误，从而阻止您连接至数据库以更改显示在 Web 站点上的信息。

要停止服务器：

1. 打开“服务器”视图。此视图通常在工作台底部的中央。如果找不到“服务器”视图，则单击主菜单中的**窗口 → 显示视图 → 服务器**。
2. 选择要停止的服务器。请注意，服务器状态列示为**已启动**。



3. 单击“服务器”视图右上方的**停止服务器按钮** 。服务器的**状态**就会更改为**正在停止**。服务器关闭之后，您将会在“控制台”视图中看到一些消息。当“服务器”视图中服务器的状态更改为**已停止**时，服务器就已停止，您可以继续处理 Web 站点。

提示：可随时在服务器上测试 Web 站点，但要记住在完成测试之后要停止服务器。

课程要点

您已经完成了课程 1.3。在本课程中，在测试环境中测试了 JSP 来模拟一个页面在实际发布时将具有的外观。还学习了如何启动和停止服务器。

课程 1.4：创建新记录以显示和更新数据库

在本课程中，将创建一个页面，该页面使用户能够创建新的分类信息广告并将它们发布至数据库。

首先，将创建关系记录来表示数据库中的新记录，然后在页面上以一组输入字段的形式来创建该记录的可视化表示。

要创建新的关系记录：

1. 在“项目资源管理器”视图中双击 **new_record.jsp** 文件以打开该文件。
2. 删除缺省文本区域的缺省内容。
3. 在“选用板”视图中，单击**数据**抽屉以展开该抽屉。
4. 将**关系记录**组件从“选用板”拖动到空白内容区域上。将打开“添加关系记录”窗口。

注：如果出现警告消息并且声明无法与数据库建立连接，则原因是在测试 Web 站点之后服务器仍在运行。如果发生这种情况，则在每个对话框中单击**取消**，并按课程 1.3 中所述停止服务器。

5. 在名称字段中，输入 `create_record`。
6. 在为以下项创建控件下面，单击**创建新的记录**。
7. 确保选择了添加输入 / 输出控件以在 **Web** 页面上显示记录。添加关系记录窗口应类似于下图：

Add Relational Record

Relational Record
Enter a reference name for record.

Name:

Create a name to refer to this record within the page.

☐ Retrieve an existing record or record list from scope

Scope :

Key :

☐ Reuse metadata definition from an existing record or record list

Input file:

You can automatically add data controls to your page to work with this record (configure details on last page of this wizard).

☒ Add input/output controls to display the record on the web page

Create controls for:

☐ Displaying an existing record (read-only)

☐ Updating an existing record

☒ Creating a new record

8. 单击下一步。
9. 在“表”框中，展开 **W5SAMPLE** 并选择 **ADS**。单击下一步。
10. 再次单击下一步。将打开配置数据控件页面。

提示：配置数据控件页面帮助您定制关系记录的可视化形式。例如，可以更改输入表单上的列、字段标签和提交按钮。在完成这些步骤之后，一个功能齐全的输入表单将出现在页面上。

11. 在**要显示的字段**部分，清除每个字段名旁边的复选框（您想要显示在输入表单中的那些字段名除外）：
 - DESCRIPTION
 - ID
 - MAINCATEGORY
 - PHONE
 - PRICE
 - TITLE
12. 通过单击“上移”或“下移”按钮来按以下顺序重新排列各个字段：
 - ID
 - TITLE
 - DESCRIPTION
 - MAINCATEGORY
 - PRICE
 - PHONE
13. 将标签重命名为您喜欢的名字。例如，将标签“Maincategory:”缩短为“Category:”。要对为各个输入字段生成的标签进行重命名，单击**标签**列中的标签。鼠标的图标将变成一个光标，以便您可以输入新文本。
14. 选择 **MAINCATEGORY**，然后单击**选项**。将打开选项窗口。
15. 确保选择了**提交按钮**选项。
16. 在**标签**字段中输入发布新列表。然后，单击**确定**。“添加关系记录”窗口现在应类似于下图：

Add Relational Record

Configure Data Controls

Specify the columns to display and how to display them

Fields to display:

Field Name	Label	Control Type
☒ ID (java.lang.Integer)	Id:	Input Field
☒ TITLE (java.lang.String)	Title:	Input Field
☒ DESCRIPTION (java.lang.String)	Description:	Input Field
☒ MAINCATEGORY (java.lang.String)	Category:	Input Field
☒ PRICE (java.math.BigDecimal)	Price:	Input Field
☒ PHONE (java.lang.String)	Phone:	Input Field
☐ TYPECODE (java.lang.String)	Typecode:	Input Field
☐ DATEOPEN (java.util.Date)	Dateopen:	Input Field
☐ CATEGORY_ID (java.lang.Integer)	Category_id:	Input Field
☐ TYPE (java.lang.String)	Type:	Input Field
☐ EMAIL (java.lang.String)	Email:	Input Field
☐ PHOTO (byte[])	Photo:	File Upload
☐ IMAGETYPE (java.lang.String)	Imagetype:	Input Field
☐ LOCATION (java.lang.String)	Location:	Input Field

17. 单击完成以生成输入表单。它应类似于下图:

The screenshot shows a web application titled "classifieds". At the top is a navigation bar with links: "Home", "View all listings", and "» Post a listing". On the left is a sidebar with the same links. The main content area contains a form titled "Post a listing »". The form has the following fields:

Id:	{ID}	123
Title:	{TITLE}	abc
Description:	{DESCRIPTION}	abc
Category:	{MAINCATEGORY}	abc
Price:	{PRICE}	123
Phone:	{PHONE}	abc

Below the fields is a section labeled "[Error Messages]" with a red icon. At the bottom of the form is a button labeled "Post New Listing".

注：该表单具有一个“错误消息”字段。这并不表示项目有错误。当用户提交表单时出现错误，此字段会标记出错位置。

18. 保存该页面。

提交输入表单之后，页面会自动将新记录添加至数据库。在下一课程中，您将对**发布新列表**按钮进行编程以返回到 `all_records.jsp` 页，以便可以立即查看数据库中的新记录。

课程要点

您已经完成了课程 1.4。在本课程中，创建了一个页面，该页面使用户能够创建新的分类信息广告并将它们发布至数据库。

课程 1.5：对提交按钮进行编程

提交输入表单之后，页面会自动将新记录添加至数据库。在本课程中，将对**发布新列表**提交按钮进行编程以返回到 `all_records.jsp` 页，以便可以立即查看数据库中的新记录。

要对提交按钮进行编程：

1. 在 `new_record.jsp` 文件中，右键单击您在 Web 页面上的表单中创建的**发布新列表**按钮，然后选择**属性**。
2. 单击**添加规则**。将打开添加导航规则窗口。
3. 从“页面”下拉菜单中，选择 **all_records.jsp**，然后单击**确定**。
4. 保存该页面。如果您想尝试将一条记录添加至数据库以验证是否将返回到 `all_records.jsp` 页面，则在测试服务器上运行 `new_record.jsp`。

防止重复键

要点：

由于 ID 列是 ADS 表中的主键，因此，不能添加具有表中已存在的 ID 值的记录。在“模块 2”中，您将了解如何使用自动键生成来为每个新记录自动创建未使用过的新键。

在此之前，您必须在此页面中输入未使用过的 ID 号来添加新记录。这些由数据库本身提供的记录使用 1 至 24 之间的某个 ID 号，因此，您可以输入大于 24 的任何数字来作为主键。如果输入了多个记录，则应确保键不重复。

将输入绑定至关系记录（可选）

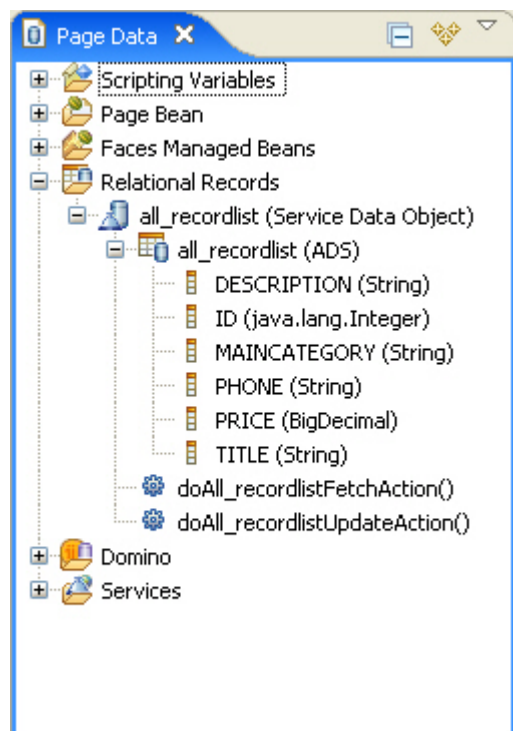
页面上的输入表单实际上是一组已绑定至所创建的关系记录的 JavaServer Faces 输入字段。回想一下，您已创建了关系记录来表示数据库的 ADS 表中的新记录。绑定是一种用来将 JavaServer Faces 输入组件链接至关系记录中的列的方法。

创建关系记录时，向导自动将该记录的所有列都绑定至页面上的输入字段。如果想要手工进行更改，则可以将其他列绑定至其他输入字段。为了将关系记录中的列绑定至输入字段，可将该列从“页数据”视图拖动到该字段上。可以通过先删除然后再重新创建表单上的“描述”输入字段来实践此过程。

更多地了解“页数据”视图：

“页数据”视图通常位于工作台左下角中。如果找不到它，则转至菜单栏并单击窗口 → 显示视图 → 页数据。通常，仅当创建动态 Web 页面时才会使用“页数据”视图。

“页数据”视图以数据对象（如关系记录或关系记录列表）形式存储与数据源（如数据库）的连接。这些数据对象不表示数据源本身或页面上的任何组件。这些数据对象表示的是项目与数据源之间的连接。一旦创建了此连接，就可以将数据对象从“页数据”视图拖动到 Web 页面上以完成连接并在 Web 页面上显示数据源中的信息。例如，下图是一个“页数据”视图，它显示了在本教程中创建的关系记录列表：



在此图中，可以看到 `all_recordlist` (ADS) 关系记录列表，下面是此关系记录列表中的所有列。此数据对象已定义并位于“页数据”视图时，您不必重新连接至数据库就可以根据需要将这些列拖动到 Web 页面上的输入组件。

例如，如果想要在 Web 页面上显示 `DESCRIPTION` 列的值，首先应将一个文本输出字段拖到该页面上，然后将 `DESCRIPTION` 列从“页数据”视图拖到该文本输出字段上。此过程称为将列绑定至输出字段。

或者，如果您想要允许用户为数据库中的 `DESCRIPTION` 列输入新值，应将一个文本输入字段拖到该页面上。然后，将 `DESCRIPTION` 列拖到该文本输入字段上。

提供了下列步骤来说明输入字段的概念和绑定过程。但是，在本教程中，是否需要完成这些步骤是可选的。如果不想完成这些步骤，则转至下一课程。

1. 在 `new_record.jsp` 文件中单击**描述**输入字段。按 **Delete** 键。
2. 在“选用板”视图中，单击**增强型 Faces 组件**抽屉以展开该抽屉。
3. 将**输入**组件从“选用板”拖动到包含刚删除的**描述**输入字段的单元格。此单元格中现在有一个输入字段，但该单元格中没有诸如 `{ID}` 或 `{TITLE}` 之类的文本，因为此输入字段未绑定至任何列。
4. 在“页数据”视图中，单击以展开**关系记录** → **create_record** (服务数据对象) → **create_record** (ADS)，并将 `DESCRIPTION` 列拖动到刚创建的输入组件上。输入组件中的文本发生了变化，指示它现在被绑定至 `DESCRIPTION` 列，如下图所示：

Id:	{ID} 123
Title:	{TITLE} abc
Description:	{DESCRIPTION} abc
Category:	{MAINCATEGORY} abc
Price:	{PRICE} 123
Phone:	{PHONE} abc

{Error Messages}

Post New Listing

5. 保存该文件，愿意的话，在测试服务器上运行该页面。

课程要点

您已经完成了课程 1.5。在本课程中，学习了如何对命令按钮进行编程以及通过将 JSF 窗口小部件从选用板拖放到 Web 页面上来将数据绑定至输入字段。

课程 1.6: 创建更新页面

此时，您已经创建了用于查看和创建 Web 站点列表的页。在此练习中，您将创建允许用户更新和删除列表的页面。更新页面与创建页面看起来几乎完全相同，不同之处是在更新页面上，输入字段将显示现有记录的数据以供用户更改。

首先，您将创建一条关系记录，它表示数据库中的一条现有记录。接下来，将为此关系记录创建 JavaServer Faces 更新表单，并且稍加更改之后，就可以测试您的页面。

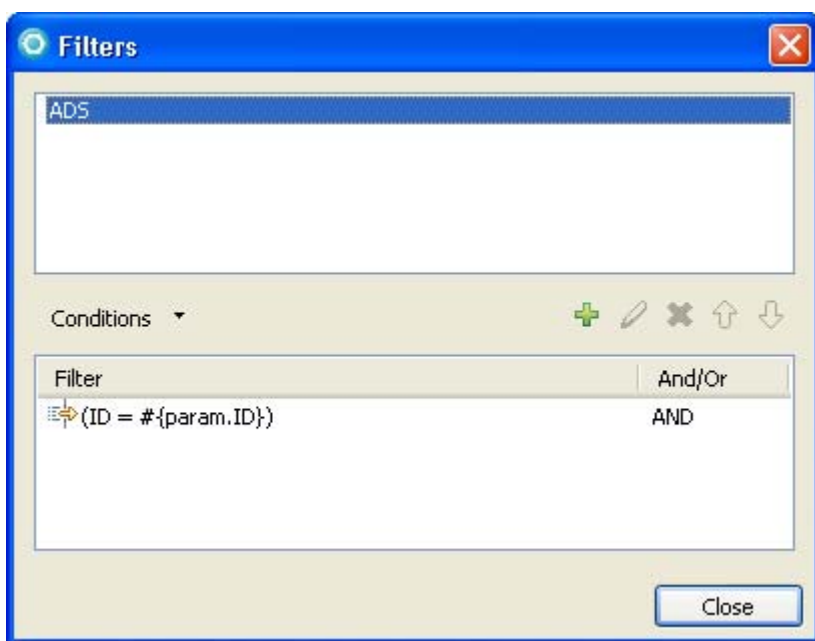
创建更新关系记录

1. 通过在“项目资源管理器”视图中双击 **update_record.jsp** 文件来打开该文件。
2. 删除缺省文本文本区域的缺省内容。
3. 在“选用板”视图中，单击**数据**抽屉以展开该抽屉。
4. 将**关系记录**组件从“选用板”拖动到空白内容区域上。将打开添加关系记录窗口。
5. 在“名称”字段中，输入 **update_record**。
6. 在为以下项创建控件下面，单击**更新现有记录**。
7. 确保选择了添加输入 / 输出控件以在 **Web** 页面上显示关系记录选项。
8. 单击下一步。
9. 在“表”框中，展开 **W5SAMPLE** 并选择 **ADS**。
10. 单击下一步。将打开“添加关系记录”页面。

对结果进行过滤

关系记录仅可显示数据库中的一条记录。因此，您必须过滤数据库表以便仅显示一条记录来供用户编辑。您不需要过滤上一个练习中的数据库，这是因为您创建了新记录，所以没有来自数据库的要过滤的结果。由于数据库中的每条记录都具有唯一的 ID 号，所以将把结果过滤为具有给定 ID 号的一条记录。

1. “添加关系记录”页面具有任务链接的列表。在**任务**下面，单击**对结果进行过滤**。将打开“过滤”窗口，并且在**过滤**列中插入缺省过滤条件 **ID = #{param.ID}**。“过滤”窗口类似于下图：



。此代码会过滤数据库中的记录，以使只有具有指定 ID 号的记录才会出现在关系记录中。您将在此练习中稍后描述的“添加行操作”一节了解到有关此条件的更多内容。单击**关闭**。

2. 单击下一步。将打开“配置数据控件”页面。
3. 在**要显示的字段**部分，清除每个字段名旁边的复选框（您想要显示在输入表单中的那些字段名除外）：
 - ID
 - TITLE
 - DESCRIPTION
 - MAINCATEGORY

- PRICE
 - PHONE
4. 通过单击上移或下移，将字段名按照自顶向下的顺序重新排序，如下所示：
- ID
 - TITLE
 - DESCRIPTION
 - MAINCATEGORY
 - PRICE
 - PHONE
5. 对于 ID 字段，从控件类型列的下拉列表中选择输出字段。虽然您希望用户能够查看记录的 ID 号，但是不希望他们能够对它进行更新。使 ID 字段成为“输出”字段将有助于避免产生重复 ID 的问题。
6. 单击选项。将打开“选项”窗口。确保选择了提交按钮选项。在标签字段中输入更新。单击确定。“添加关系记录”窗口应类似于下图：

Add Relational Record

Configure Data Controls

Specify the columns to display and how to display them

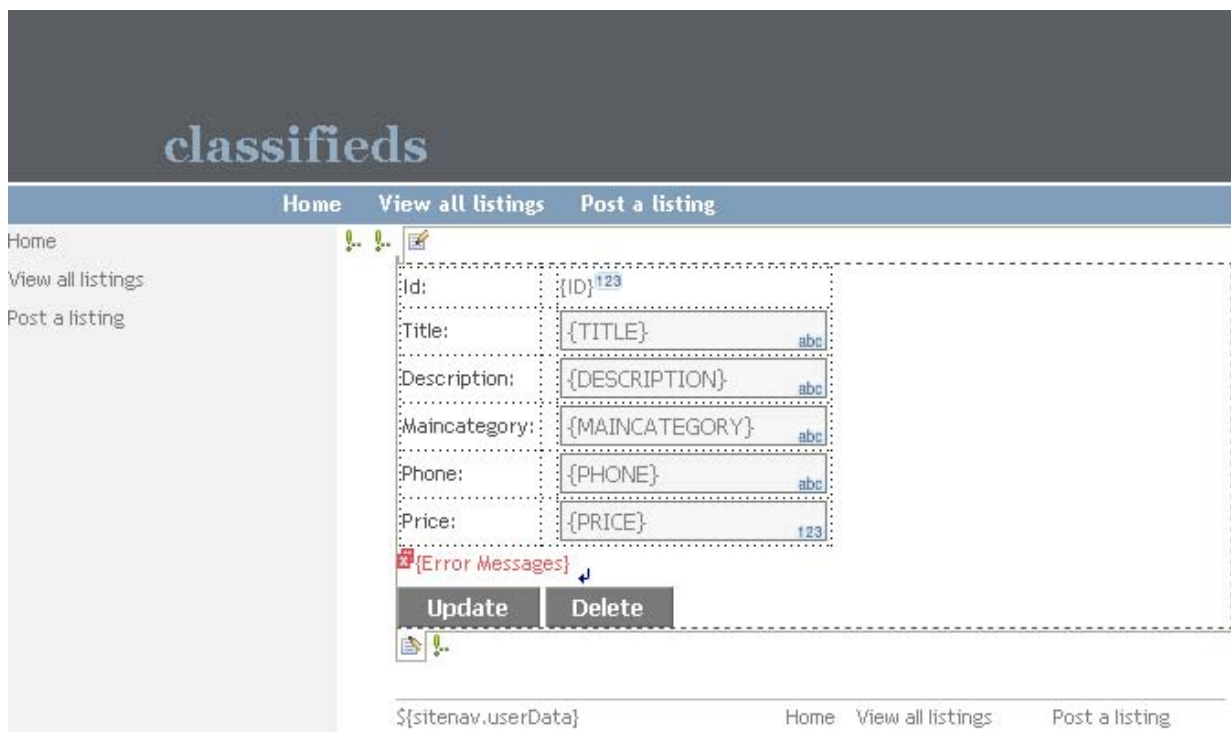
Fields to display:

Field Name	Label	Control Type
☒ ID (java.lang.Integer)	Id:	Output Field
☒ TITLE (java.lang.String)	Title:	Input Field
☒ DESCRIPTION (java.lang.String)	Description:	Input Field
☒ MAINCATEGORY (java.lang.String)	Maincategory:	Input Field
☒ PRICE (java.math.BigDecimal)	Price:	Input Field
☒ PHONE (java.lang.String)	Phone:	Input Field
☐ TYPECODE (java.lang.Short)	Typecode:	Input Field
☐ DATEOPEN (java.util.Date)	Dateopen:	Input Field
☐ CATEGORY_ID (java.lang.Integer)	Category_id:	Input Field
☐ TYPE (java.lang.String)	Type:	Input Field
☐ EMAIL (java.lang.String)	Email:	Input Field
☐ PHOTO (byte[])	Photo:	File Upload
☐ IMAGETYPE (java.lang.String)	Imagetype:	Input Field
☐ LOCATION (java.lang.String)	Location:	Input Field

Buttons: All, None, Options...

Navigation: ? < Back, Next >, Finish, Cancel

7. 单击完成以在页面上生成更新表单，如下图所示：



对“更新”和“删除”按钮进行编程

此外，将添加代码以使用户能够查阅 `all_records.jsp` 页来显示更改过的记录以及所有其他记录。

1. 右键单击刚才在 Web 页面上创建的更新按钮，然后选择属性。
2. 单击添加规则。将打开添加导航规则窗口。
3. 从“页面”下拉菜单中，选择 **all_records.jsp**。在此规则的适用对象框中，单击仅适用于此操作。确保该字段包含：`#{pc_Update_record.doUpdate_recordUpdateAction}`。单击确定。
4. 重复步骤 1 到步骤 3 以在 Web 页面上按相同的方式为“删除”按钮编辑代码。确保此规则的适用对象字段中包含：`#{pc_Update_record.doUpdate_recordDeleteAction}`
5. 保存该页面。如果您想尝试将一条记录添加至数据库以验证是否将返回到 `all_records.jsp` 页面，则在测试服务器上运行 `new_record.jsp`。

添加行操作

现在，您将对 `all_records.jsp` 上的表添加行操作，以使用户可选择要更新的数据库记录。`#{param.ID}` 表示更新页面将要更新的记录的 ID 号。当用户单击某一行时，该记录的 ID 号将被发送至 `update_record.jsp` 页作为 `#{param.ID}` 参数。然后，刚才插入到 `update_record.jsp` 页中的已过滤关系记录将仅显示该记录以用于更新。

1. 在“项目资源管理器”中，双击 **all_records.jsp** 文件以在编辑器中打开该文件。单击数据表中的任何位置。打开“属性”视图。
2. 在“属性”视图中，单击 HTML 标记列表中的 **hx:dataTableEx** 下面的行操作选项卡。
3. 单击添加单击行时就会执行的操作。将打开“配置行操作”对话框。选择单击行时就会对服务器发送请求，并将行信息作为参数来发送，然后单击确定。这就会将一行添加至数据表。
4. 在“属性”视图中单击 **hx:requestRowAction** 选项卡。
5. 单击添加规则。将打开“添加导航规则”对话框。

6. 从“页面”下拉菜单中，选择 **update_records.jsp**。在“此规则的适用对象”框中，选择**仅适用于此操作**。会自动向“仅适用于此操作”字段中填充您在添加行操作时所创建的 `#{pc_All_records.doRowAction1Action}`。单击**确定**。如果没有为该字段自动填充以上方法，则在该字段中输入此方法，并对“ClassifiedsTutorial / Java 资源”添加以下内容：src/pagecode/All_records.java:

```
public String doRowAction1Action() {  
    return "";
```

7. 在“属性”视图中单击 `hx:requestRowAction` 下的**参数**选项卡。单击**添加参数**，然后在“名称”字段中输入 `ID`。需要将 `ID` 参数绑定至 `ID` 列。将行操作绑定至数据列表的 `ID` 列意味着单击行时，请求参数将是列表中的记录的 `ID`。
- 单击**值**，然后单击**选择页数据对象**  按钮。
 - 在**数据对象**选项卡中，展开 **all_recordlist (服务数据对象)** → **all_recordlist (ADS)**，然后选择 **ID (java.lang.Integer)**。
 - 单击**确定**。

现在，当用户单击一行时，Web 站点将允许用户更新有关分类信息广告的信息。保存该文件，愿意的话，测试该页面。请记住，要先打开 `all_records.jsp`，因为这是链接至 `update_record.jsp` 的页面。

课程要点

您已经完成了课程 1.6。在本课程中，您学习了如何创建可用来更新数据库中的记录的 Web 页面。

模块 1: 总结

您已经完成了“模块 1: 创建具有数据连接的 Web 页面”。

已学习的课程

现在，您已了解如何完成下列任务：

- 在 Web 页面上显示数据库中的信息。
- 使用关系记录、关系记录列表和数据表。
- 在 Web 页面中显示、编辑、创建和删除数据库记录。
- 通过在数据表组件上使用行操作链接一条或一组记录。

此模块向您介绍了 JavaServer Faces 技术和工具。您还可以执行更多的任务来使您的站点更具吸引力、更可用且效率更高。在下一个模块（即，第 25 页的『模块 2: 添加高级功能』）中，您将学习如何使用这个简单的分类信息广告站点并使它成为一个更复杂而且视觉效果更好的应用程序。如果您想了解如何使您的站点类似于下图，请继续学习下一个模块：

classifieds

[Home](#)
[» View all listings](#)
[Post a listing](#)

[Home](#)
[View all listings »](#)
[Post a listing](#)

View all listings

Title	Details	Category
WV Passat wagon 	Description: 2000 good condition Price: \$1,000.00 Phone: 416-512-8665	Automotive
Magic Flute 	Description: Imagine it. Price: \$101.00 Phone: 314-159-265	Entertainment
Video Camera 	Description: Great camera, suitable for amateur or professional use. Price: \$900.00 Phone: 534-333-454	Entertainment
VCR 	Description: Old and worn but still working ok, hence low price. Price: \$40.00 Phone: 322-654-044	Entertainment
Flat Screen TV 	Description: Very clear picture, sound excellent too. Reluctant sale. Price: \$700.00 Phone: 456-456-944	Entertainment

1 2 3 4 5

模块 2: 添加高级功能

在此模块中，将学习如何使用高级功能来修改 Web 页面的格式、添加使用户能够将文件上载至数据库的功能、设置会对用户显示特定页面的规则，以及实现自动完成一些诸如生成密钥之类的管理任务。

学习目标

此模块将介绍一些功能更强大的方法来使用数据库中的数据。在本模块中，您将完成下列任务：

- 格式化 Web 页面上的数据库记录
- 添加一个组件，它允许您将文件从 Web 页面上载到数据库中
- 从一个页面自动浏览至另一个页面
- 自动执行某些数据库管理任务

所需时间

完成此模块大约需要 1 小时 30 分钟。如果您决定在学习本教程的同时研究动态 Web 站点的其他方面，则可能要花更长时间才能完成本教程。

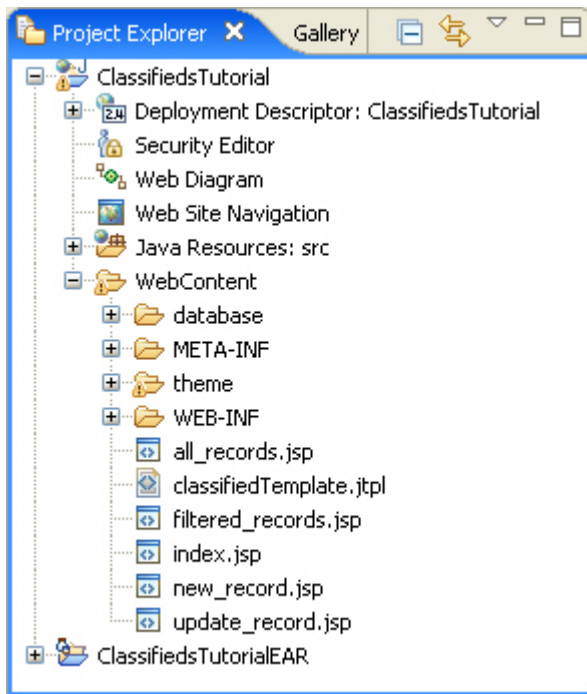
先决条件

如果您已经完成了“模块 1：创建具有数据连接的 Web 页面”，则跳过此“先决条件”部分的其余步骤，并开始学习第 29 页的『课程 2.1：格式化数据表』。

如果您未首先完成模块 1 而直接开始完成本教程中的模块 2，则必须首先导入必需的资源、设置目标服务器并建立数据库连接。

要导入必需的资源：

1. 导入项目。切换至 Web 透视图（窗口 → 打开透视图 → **Web**）。
2. 在 Web 透视图的“项目资源管理器”视图中，确保 ClassifiedsTutorial 项目类似于下图：



要设置目标服务器：

1. 在 Web 透视图的“项目资源管理器”视图中，右键单击 **ClassifiedsTutorial** 并选择**属性**。
2. 在属性列表中，单击**服务器**。
3. 在“缺省服务器”列表中，选择想要使用的缺省服务器。单击**应用**。
4. 在属性列表中，单击**目标运行时**。
5. 在“运行时”列表中，单击与您所选择的服务器相对应的运行时。单击**确定**。

如果目标运行时列表中未列示您想使用的服务器，则关闭工作台并安装想要使用的服务器。一旦安装了该服务器，则遵循指示信息来设置目标服务器。

注： 如果未看到“缺省服务器”列表中列示任何服务器，但是您已经安装了服务器运行时，则可能需要配置服务器。要配置服务器，可以执行下列操作：

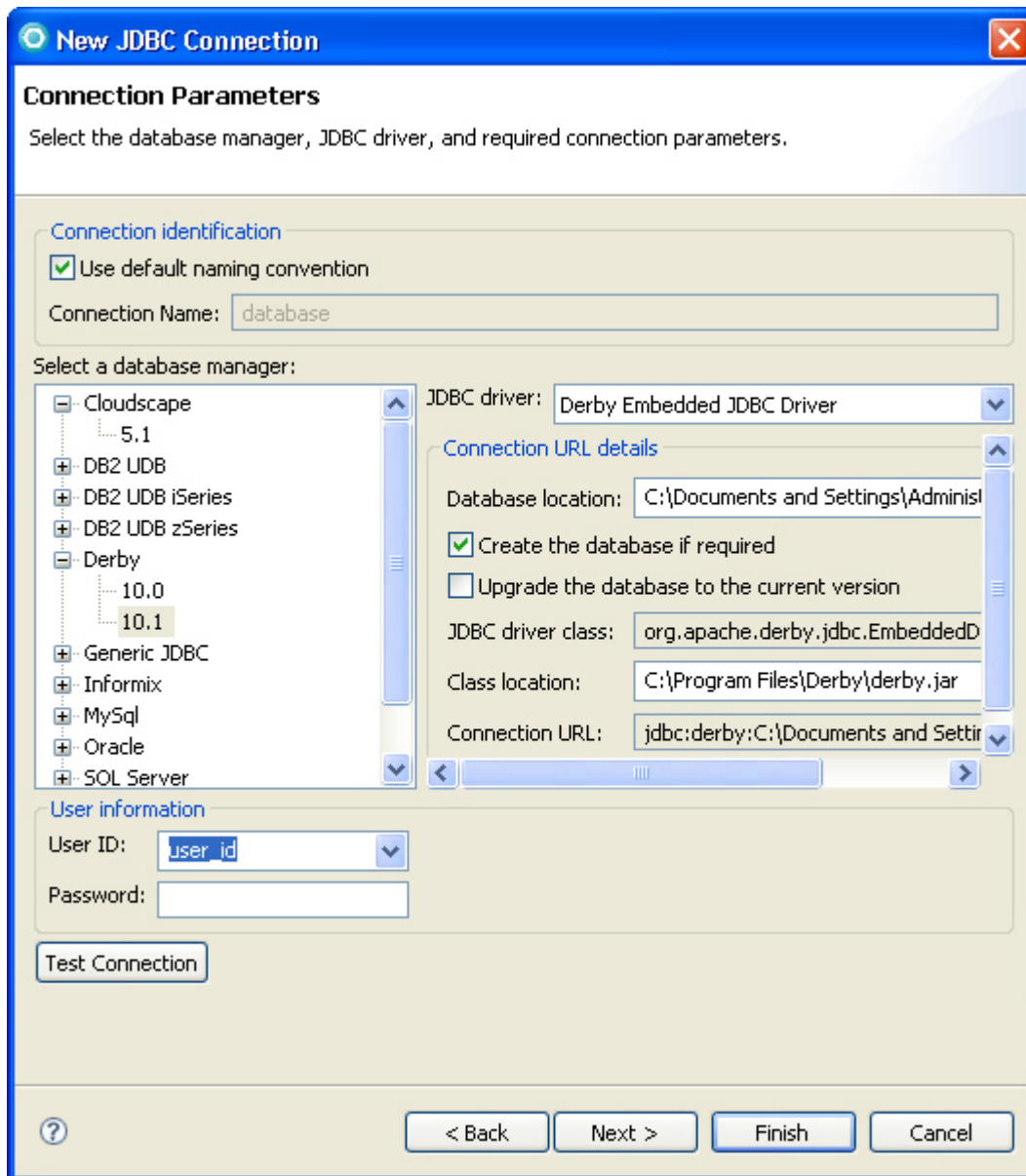
1. 右键单击 **all_records.jsp** 文件，然后单击**运行方式** → **在服务器上运行**。
2. 选择**手工定义新的服务器**。
3. 选择已安装的服务器。
4. 遵循向导中的指示来配置服务器。当您首次在服务器上运行时，可能会接收到错误。要修正该错误，按以上所述设置目标服务器，在“服务器”视图中重新启动服务器，并在浏览器中重新装入 Web 页面。

如果重新执行前面所描述的步骤来设置目标服务器，则会发现缺省服务器就是您刚才所配置的服务器。如果已经安装但是未配置服务器，则该服务器将不会显示在服务器列表中，您可以从该列表中选择缺省目标服务器。

有关在 WebSphere Application Server 之外的服务器上部署 SDO 的信息，请参阅帮助主题：在非 WAS 服务器上部署 SDO。

要建立数据库连接：

1. 在“项目资源管理器”视图中，右键单击 **ClassifiedsTutorial** 并选择**属性**。将打开“ClassifiedsTutorial 的属性”窗口。
2. 单击 JDBC 连接。
3. 在“JDBC 连接”属性中，单击**新建**。“新建 JDBC 连接”窗口打开。
4. 单击**创建新的连接**，然后单击**下一步**。
5. 在**选择数据库管理器**列表中，展开 **Derby**，然后选择 **10.1**。
6. 在“数据库位置”字段中，单击**浏览**并选择 `<workspace_location>\ClassifiedsTutorial\WebContent\database`，其中 `<workspace_location>` 是当前工作空间的目录。单击**确定**以关闭浏览以找到文件夹窗口。
7. 在**类位置**字段中，输入用来实现 JDBC 驱动程序类的一个或多个 JAR 所在的位置，或者单击**浏览**以选择文件系统中的某个位置。选择了 JAR 文件的位置之后，单击**打开**。
8. 可能需要输入用户标识才能访问数据库。但密码不是必需的。“新建数据库连接”向导应类似于下图：



9. 在“新建 JDBC 连接”向导中，单击**完成**。您将返回到 ClassifiedsTutorial 的属性窗口。
10. 单击**配置项目的数据库连接**。将打开“JDBC 连接属性”窗口。
11. 单击“运行时连接详细信息”部分旁边的**编辑**。选择**使用驱动程序管理器连接**作为运行时连接类型。单击**完成**，然后单击**确定**。

可以浏览教程 Web 项目中的文件。要打开某个文件，在“项目资源管理器”视图中双击该文件。要查看用类似站点图表示的页面之间的相关性，双击“项目资源管理器”中的 Web 网站导航。

您在此样本中的主要工作将涉及到下列文件：

all_records.jsp

这是站点的主页。它将显示数据库中的每一条分类信息广告。

new_record.jsp

此页面将创建新的分类信息广告。

update_record.jsp

此页面将更改有关数据库中广告的详细信息或者删除广告。

classifiedTemplate.jtpl

这是站点页面的模板。它包括诸如出现在每个页面上的表和“欢迎使用分类信息广告”灰色条幅之类的元素。此页面在灰色条幅下方还有两个导航标签，分别引导至主页和新的分类信息广告页面。

课程 2.1: 格式化数据表

all_records.jsp 页上的数据表具有实用的外观。它显示数据库中的所有记录以及指向它打算链接至的其他页面的链接，但外观不是很吸引人。在此练习中，将通过添加分页控件、样式规则和待售商品的图片来改善此数据表。

将列标题重命名

很多情况下，您可能不希望数据表中的列标题正好就是数据库中的列名。下列步骤将说明如何将数据表列标题重命名为更适当的名称。

1. 在“项目资源管理器”视图中双击 **all_records.jsp** 页。
2. 单击数据表中的 **Maincategory** 列标题。
3. 在“属性”视图中，单击 **h:outputText** 选项卡并将值文本字段从 **Maincategory** 更改为 **Category**。
4. 如果您愿意，可以重命名其他列。
5. 保存该页面。

格式化输出组件

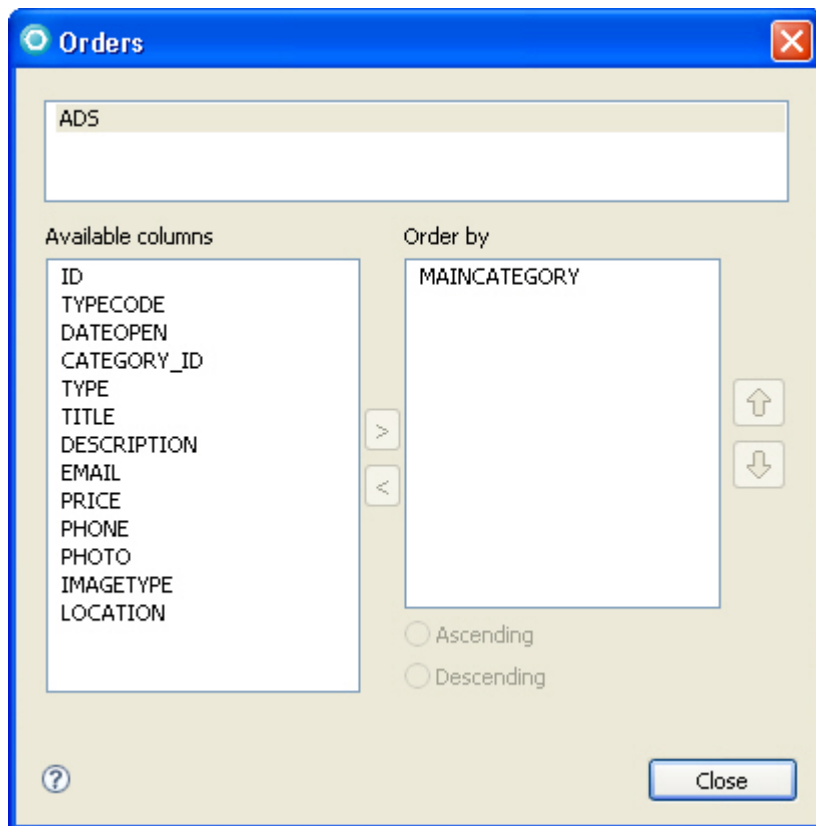
还可以格式化输出组件。在这些步骤中，将格式化 {PRICE} 输出组件，使之以货币值（而不是普通的数字）的形式出现。

1. 在页面上单击 {PRICE} 输出组件。
2. 在“属性”视图中，查找类型下拉列表，并选择货币。现在，每件商品的价格都将以货币样式显示，而不是显示为普通数字。其他样式和格式（如百分比、日期和时间）可用于不同类型的数据。还可以定制输出组件的格式以显示未列示在“属性”视图中的某种类型的数据（如电话号码）。
3. 保存该页面。

对数据进行排序

现在，数据未按任何顺序列示。为了向用户提供一个更有条理的广告列表，按类别对记录列表进行排序，以便类似的商品都集中在一起。

1. 在“页数据”视图（窗口 → 显示视图 → 页数据）中，右键单击 **all_recordlist (ADS)** 并单击上下文菜单中的配置。将打开配置关系记录列表窗口。如果出现警告消息并且声明无法与数据库建立连接，则原因是在测试 Web 站点之后服务器仍在运行。如果发生这种情况，则在每个对话框中单击“取消”，并按测试 Web 站点中的“停止服务器”部分中所述停止服务器。
2. 选择从作用域中检索现有记录或记录列表和复用现有记录或记录列表中的元数据定义，然后单击下一步。
3. 在下一个窗口中，单击以选择 ADS，然后单击下一步。
4. 在下一个窗口的右边，单击对结果进行排序。这将打开“排序”窗口。
5. 在“可用的列”窗格中，单击 MAINCATEGORY。
6. 单击“>”按钮以将 MAINCATEGORY 列移到“排序方式”窗格中。该窗口应类似于下图：



7. 单击关闭以关闭“排序”窗口。
8. 单击完成以应用更改。现在，数据将按类别排序。可以按数据库中的任何列排序。

添加分页控件

为了使每条记录不会同时显示在数据表中，可以使用分页控件。分页控件自动将记录分到设定大小的多个页中，而不必在项目中创建任何新的 JSP 文件。

1. 单击数据表中的任何位置。
2. 在“属性”视图中，单击视图左边的 HTML 标记列表中的 **h:dataTableEx**。
3. 单击 **h:dataTableEx** 下面的显示选项选项卡。
4. 在每页行数字段中，输入 5。
5. 单击添加 **Web 样式分页控件** 旁边的按钮。在数据表的底部将出现分页控件的大致外观的图形。Page Designer 中的分页控件的可视化形式是一个占位符图像，并不真正反映将显示的页数，这是因为只有在浏览器中装入页面时才能确定页数。
6. 保存该页面。如果您愿意，可以使用不同样式的分页控件和页面统计信息组件来进行实践，从而找出最适用于该页面的分页控件和页面统计信息组件。

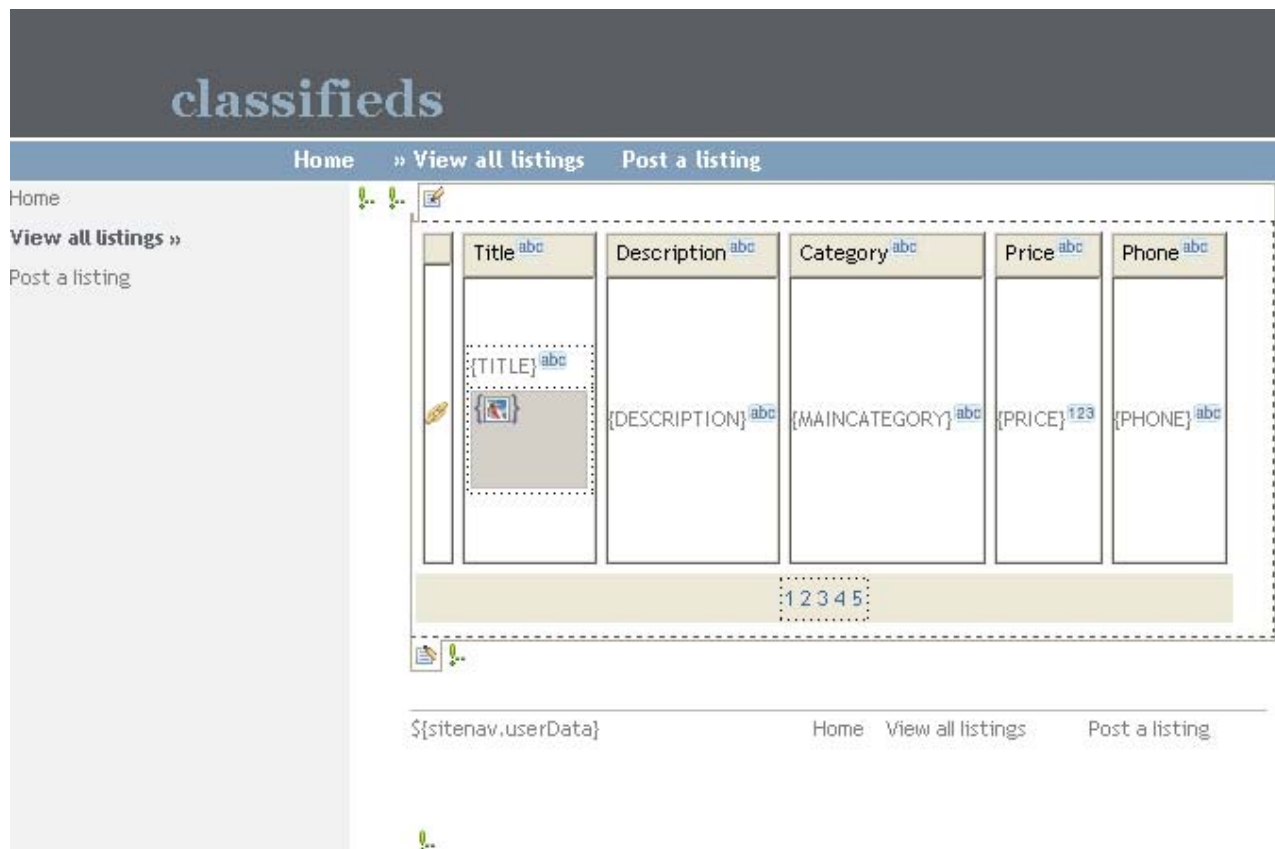
以列表形式对组框中的组件进行布局

不需要让数据表的每一列都有一个输出组件，在列中组合多个元素就可以获得更美观的布局。您将使用组框来对各组件进行排列，与使用隐藏的 HTML 表的方式类似。具体地说，在添加图像组件以增强站点的视觉效果之后，您将获取有关每条广告的大部分文本式信息并将它放置到单个数据表列中，然后将该列标记为 DETAILS。

1. 单击“选用板”中的“增强型 Faces 组件”抽屉以打开该抽屉。将**面板 - 组框**组件拖至数据表的 TITLE 列。将打开“选择类型”窗口。

2. 在“选择类型”窗口中选择**框**作为组框类型，然后单击**确定**。此组框以单列或单行的形式来布置组框内的组件，并且可以按垂直方向或水平方向排列。
3. 单击组框以选择该组框。
4. 通过使用“属性”视图，将**方向**更改为**垂直**。
5. 将 **{TITLE}** 组件拖到列表组框中。一旦将组件添加至组框，组框中的指示性文本就会消失。
6. 将**图像**组件从“选用板”的“增强型 Faces 组件”抽屉拖放到列表组框中。它有助于将图像组件放在组框的底部边缘上以确保图像将放置在标题下方。
7. 单击刚刚添加的图像组件。
8. 使用“属性”视图的**大小**区域来将**宽度**设置为 60 像素，并将**高度**设置为 50 像素。这样可确保无论图像在数据库中的大小如何，它们始终将以 60x50 显示在页面上。
9. 通过将 **all_recordlist** 的 **PHOTO** 列从“页数据”视图中拖动到图像组件上来将图像组件绑定至该 **PHOTO** 列。这将使图像组件显示在每条记录的 **PHOTO** 列中找到的图像数据。如果未列示 **PHOTO (ByteArray)** 列，则在“页数据”视图中，右键单击 **all_recordlist (ADS)** 并从菜单中选择**配置**。如果“URL”窗口已打开，则单击**确定**以关闭该窗口。在“配置关系记录列表”窗口中，单击**下一步**。选择 **ADS**，然后单击**下一步**。单击以选中 **PHOTO (ByteArray)** 旁边的框，然后单击**完成**。
10. 单击已插入的图形。在“属性”视图的 **hx:graphicImageEx** 选项卡上，单击**类型**字段旁边的**选择页数据对象**按钮 。将打开“选择页数据对象”窗口。
11. 单击以展开 **all_recordlist (服务数据对象) → all_recordlist (ADS)**。
12. 单击 **IMAGETYPE (String)**。
13. 单击**确定**，然后保存该页面。

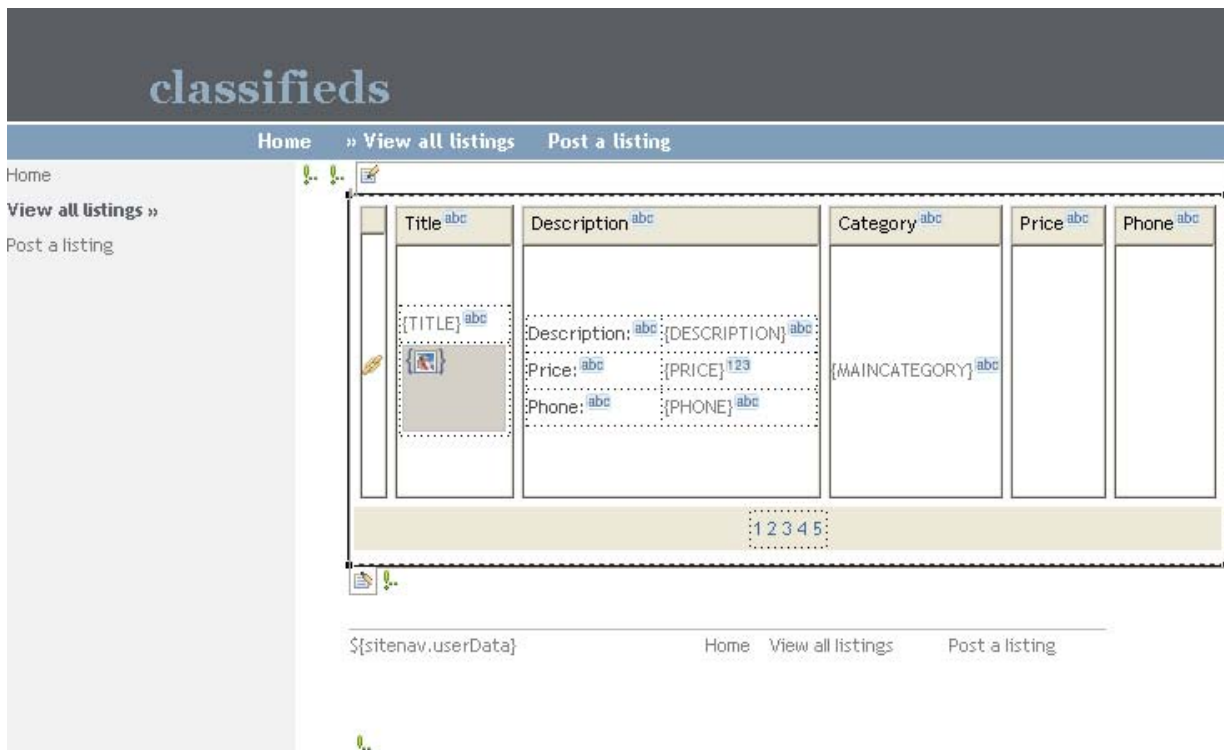
您的页面现在应该类似于下图：



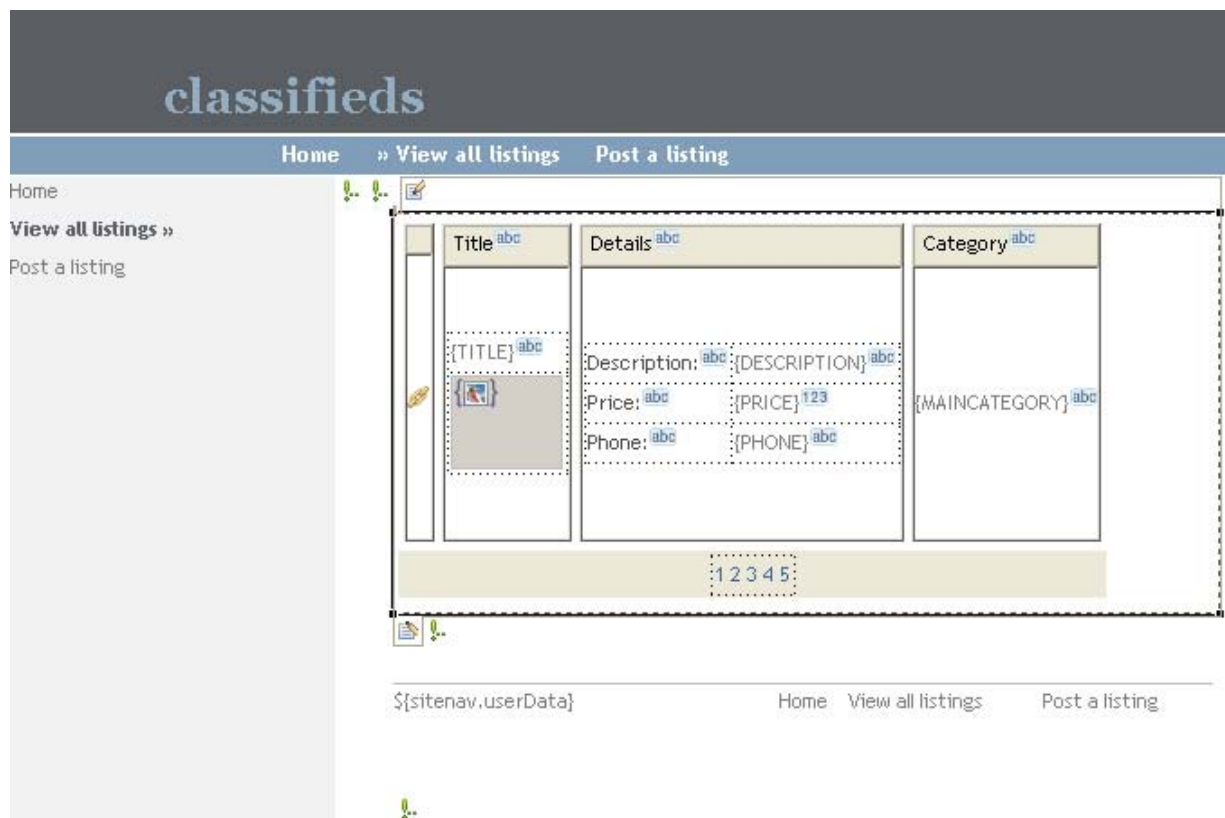
使用组框网格来优化布局

组框还可用于类似于表的格式排列组件。列表类型组框只可具有一行或一列，但网格类型组框可具有任意数目的行和列。在下面的步骤中，将把 {PRICE} 和 {PHONE} 组件及其标签移动到数据表的 DESCRIPTION 列中。

1. 将**面板 - 组框**组件从“选用板”的“增强型 Faces 组件”抽屉中拖至数据表的 DESCRIPTION 列。将打开“选择类型”窗口。
2. 单击**网格**作为要添加的组件类型并单击**确定**。
3. 单击“新建网格”组框。
4. 使用“属性”视图来将此组框的**列数**设置为 2。
5. 将**输出**组件从“选用板”的“Faces 组件”抽屉拖放到网格组框中。此输出组件将是用于描述待售商品的标签。每个输出组件将具有类似表左边单元格中的那个标签。
6. 单击**输出**组件（它在 Page Designer 中显示为**输出文本**）并使用“属性”视图将它的**值**指定为描述：。
7. 将现有 {DESCRIPTION} 组件拖到网格组框的底部边缘。如果将组件拖动到组框的右边位置时遇到问题，则尝试按下鼠标按钮并观察光标。光标在组框中的位置指示当释放鼠标按钮时组件将出现的位置。对于本练习，应在光标位于前一个组件的右边时释放鼠标按钮。
8. 将另一个**输出**组件从“选用板”拖放到组框的右边。
9. 使用“属性”视图将它的**值**指定为价格：。
10. 将现有 {PRICE} 输出组件拖放到“价格”输出文本的右边。
11. 以上述同样方式拖动 {PHONE} 组件标签的新**输出**组件，然后将它标记为电话号码：。
12. 将现有 {PHONE} 组件拖到“电话号码”输出文本右边的网格组框中。您的页面现在应该类似于下图：



13. 删除数据表中空的 PRICE 和 PHONE 列。要删除某列，单击该列并打开“属性”视图。接着，单击视图左边的 HTML 标记列表中的 **h:dataTable**，从视图右边的列表中选择要删除的列，然后单击“除去”。
14. 单击“描述”列的标题并使用“属性”视图将它的**值**字段更改为详细信息。该页面应类似于下图：



15. 保存该页面并对它进行测试。如果您不熟悉在服务器上运行应用程序，请参阅第 13 页的『课程 1.3: 测试 Web 站点』。

当测试该页面时，它应类似于下图：

classifieds		
Home » View all listings Post a listing		
Home View all listings » Post a listing		
View all listings		
Title	Details	Category
WV Passat wagon 	Description: 2000 good condition Price: \$1,000.00 Phone: 416-512-8665	Automotive
Magic Flute 	Description: Imagine it. Price: \$101.00 Phone: 314-159-265	Entertainment
Video Camera 	Description: Great camera, suitable for amateur or professional use. Price: \$900.00 Phone: 534-333-454	Entertainment
VCR 	Description: Old and worn but still working ok, hence low price. Price: \$40.00 Phone: 322-654-044	Entertainment
Flat Screen TV 	Description: Very clear picture, sound excellent too. Reluctant sale. Price: \$700.00 Phone: 456-456-944	Entertainment
1 2 3 4 5		

任何没有相关图像的广告都将显示一个断开的链接。在下一课程中，将学习如何将文件上传组件添加至页面以添加或更改广告的图像。

课程要点

您已经完成了课程 2.1。在本课程中，您已经学习了如何使用各种方法来格式化数据表。

您学习了如何完成下列任务：

- 更改列标题。
- 对表中的数据进行排序。
- 添加分页控件以将数据分成多个较小的 Web 页面。
- 使用组框并以多种方式来更改表在 Web 页面上的布局。

练习 2.2: 使用文件上传组件

通过本课程学习如何使用户能够上传分类信息广告的图片。这项功能对于 `new_record.jsp` 页和 `update_record.jsp` 页特别重要。文件上传组件使用户能够浏览他们的文件系统、将文件上传到数据库，并让此文件即时出现在数据库中。

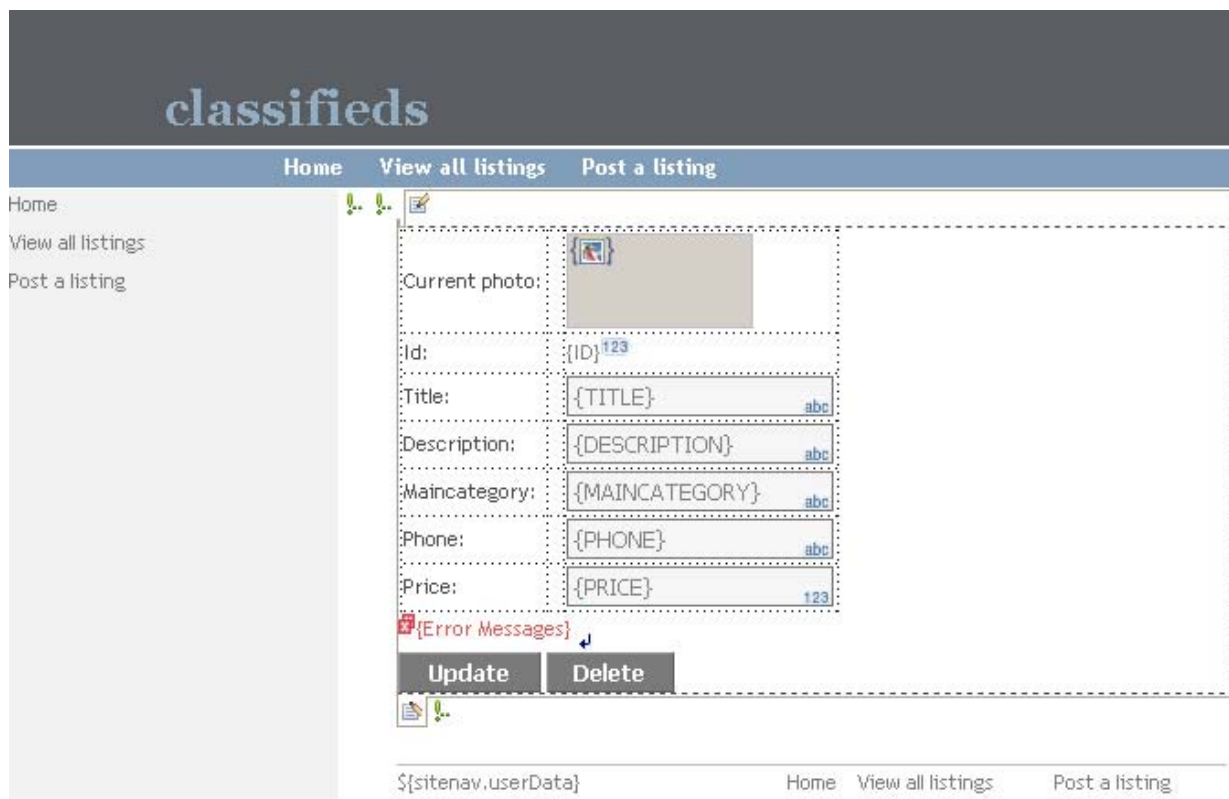
由于在创建和更新情况下使用文件上传组件的过程是类似的，所以此练习将允许更新页面对任何列表的当前图像进行更改。如果您愿意，在对更新页面完成操作时，可以对新的记录页面进行同样的更改。

将当前图片添加至页面

您将注意到，更新表单只不过是一个 HTML 表，它的静态文本和输入组件绑定至 `update_record` 关系记录中的列。知道了这一点，您就可以添加行和列来修改该表单，就像修改任何 HTML 表那样。在以下的步骤中，您将添加一个新行来显示当前图片。

在显示关系记录列表的数据表中添加行和列比较复杂。有关将列添加至数据表的信息，参阅第 6 页的『练习 1.2: 连接至数据库和在 Web 页面上显示数据』。

1. 在“项目资源管理器”视图中双击 `update_record.jsp` 页。
2. 将光标放在位于表的左上角的第一个单元格中。此单元格被标记为 ID。
3. 单击主菜单中的 **表 → 指定并添加**。将打开“添加行或列”对话框。单击 **行**，然后单击在所选单元格上方。这将在表的顶部创建一个新行，该行将包含记录的当前图片。单击 **确定**。
4. 在此新行左边的单元格中，输入当前图片：作为标签。
5. 将 **图像** 组件从“选用板”的“Faces 组件”抽屉拖动到该新行最右边的单元格中。与前一个示例不同，您将允许用户看到整个图像而不必限制图像大小，因此不要更改“属性”视图中的宽度和高度。
6. 通过将 **PHOTO** 列从“页数据”视图拖动到新的图像组件上，将图像组件绑定至 `update_record` 的 **PHOTO** 列。现在已经将图像组件绑定至数据库的 **PHOTO** 列。
7. 在仍然选择了图形的情况下，转至“属性”视图并单击“类型”字段旁边的“选择页数据对象”按钮。将打开“选择页数据对象”窗口。
8. 单击 **update_record (服务数据对象) → update_record (ADS) → IMAGETYPE (String)**。
9. 单击 **确定**。现在，如果有分类信息广告的当前图片，该页面就会将它显示出来。该页面应类似于下图：



将文件上传组件添加至页面

接下来，在表的底部添加一个新行，用来包含文件上传组件。

1. 单击最后一行，从而使光标位于该行中。
2. 从菜单栏中单击表 → 在下面添加行。
3. 在新行的第一个单元格内输入新图片：作为标签。
4. 将文件上传组件从“选用板”的“增强型 Faces 组件”抽屉中拖动到新行的最后一个单元格中。
5. 按照绑定图像组件的方式将“文件上传”组件绑定至 `update_record` 关系记录的 **PHOTO** 列。**PHOTO** 出现在文本字段中，显示此组件被绑定至 **PHOTO** 列，并且上传的文件将放在该列中。该页面应类似于下图：



6. 如果希望的话，保存该页面并对它进行测试。

课程要点

您已经完成了课程 2.2。在本课程中，您学习了如何处理更新页面上的表、将当前图片添加至表，以及将文件上下载组件添加至表。

课程 2.3: 使用导航规则

在此课程中，将学习如何设置导航规则以将站点用户重定向至特定 Web 页面。

鉴于设置 `new_record.jsp` 页的方式，用户一定要非常细心，不要输入已在使用中的 ID 号，这是因为数据库中的每条记录必须具有唯一的 ID 号。在第 18 页的『课程 1.5: 对提交按钮进行编程』中的“防止重复键”注释中对此进行了更详细的说明。在此课程中，将检查输入的 ID 是否是唯一的，如果不是，则对用户显示错误页，该错误页将描述问题并告诉用户如何解决问题。

导航规则使您在检查用户是否输入了重复的 ID 号之后，能够对用户显示错误页或 `all_records.jsp` 页。您将为这两种可能的结果指定“别名”，然后将这两个别名链接至正确的目标页面。在本例中，`create_record.jsp` 页上的错误将指定 `ERROR_CREATE` 别名，该别名将把用户引导至错误页。如果用户正确填写了 `create_record.jsp` 页，该页将指定 `MAIN` 别名，通常该别名将链接至 `all_records.jsp` 页。

设置规则

1. 在“项目导航器”中双击 `new_record.jsp` 页。
2. 单击页面上的发布新列表按钮。

3. 在“属性”视图中，单击**添加规则**。将打开“添加导航规则”窗口。如果在创建新记录并将它落实到数据库时发生某些错误，则第一条规则将对用户显示名为 `create_error.jsp` 的错误页。
4. 在**页面**字段中，输入 `error_create.jsp`。此页面不存在，但是就本教程而言，您可假定它是存在的。
5. 单击**命名结果**单选按钮。
6. 在**命名结果**单选按钮后面的文本字段中输入 `ERROR_CREATE`。
7. 单击**仅限于此页**单选按钮，因为此站点中不存在用户可通过输入重复的 ID 号来触发此特定错误的其他页。单击**确定**。如果用户输入了有效的 ID 号，则下一条规则将浏览到 `all_records.jsp`。
8. 单击**添加规则**按钮以再次打开“添加导航规则”窗口。
9. 使用页下拉框来选择 **`all_records.jsp`**。
10. 单击**命名结果**复选框然后将 `MAIN` 输入到复选框后面的文本字段中。
11. 由于您可能要在另一页（例如，`update_record.jsp` 页）中复用此规则，所以单击**此规则适用范围**下面的**所有页**单选按钮。单击**确定**。现在，这两个规则显示在“属性”视图上。

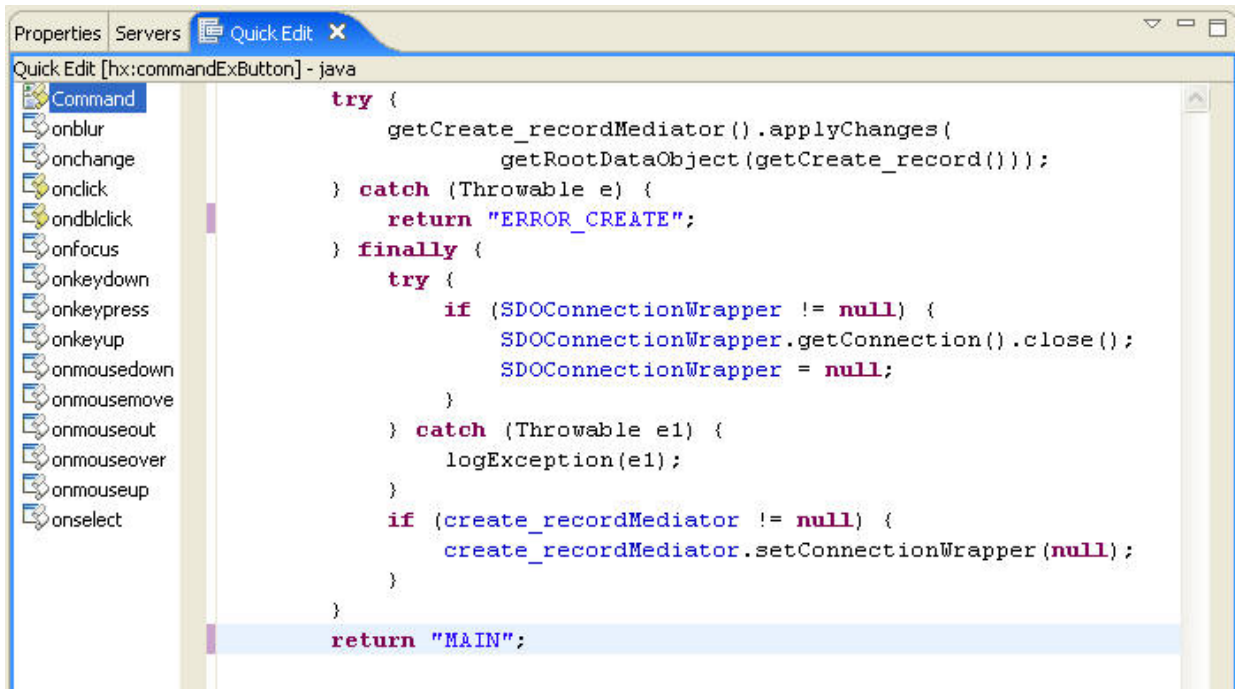
从按钮操作返回别名

现在余下的所有工作就是让新的导航规则生效。将为“发布新列表”按钮在代码中添加两个 `return` 语句。这些 `return` 语句将向适当的别名发出指示，以将用户引导至在导航规则中设置的适当页。

1. 单击**发布新列表**按钮。
2. 打开**快速编辑**视图。
3. 在“快速编辑”视图中，查找内容如下的行：`} catch (Throwable e) {`。如果用户输入了重复的 ID 号，此 `catch` 函数就将运行。
4. 除去此行末尾的左花括号 `{` 与数行后的下一个与之匹配的右花括号 `}` 之间的所有代码。不要除去任何一个花括号。
5. 在花括号之间代码的位置处输入以下文本：`return "ERROR_CREATE";`
6. 接着，除去最后一个右花括号 `}` 下面的所有代码，并在此处输入以下文本：`return "MAIN";`

此步骤会除去在练习 1.4 中添加的 `gotoPage` 操作。由于导航规则执行了同样的操作，所以不再需要此代码。

按钮的代码现在应该类似于下图：



7. 保存该页面，如果还想测试它，则可以进行测试。

（可选）可创建名为 `create_error.jsp` 的一个简单错误页，该错误页向用户说明创建他们的列表时发生了错误，他们应尝试使用另一个 ID 值。然后，您可以通过尝试添加使用现有 ID 值（例如，1）的新列表来测试这些导航规则。

课程要点

您已经完成了课程 2.3。在本课程中，学习了如何使用导航规则来实现在用户试图输入无效 ID 值就会产生错误，而在用户输入有效 ID 时就对该用户显示 `all_recordlist.jsp` 页。

课程 2.4: 使用自动键生成

在上一课程中，设置了导航规则以确保用户为新的分类信息广告输入唯一标识。因为您并不希望使用 Web 站点的用户通过反复试验而窃取唯一标识，所以此过程是无益的并且不切实际。在本课程中，将设置自动键生成以便数据库对它包含的每条新记录自动给定一个唯一编号。

自动键生成是一个复杂的主题，但是简单说来，如果某一数据库为键生成保留了一个特殊的表，该数据库就可以选择新键。此表的某一列（增量器列）必须具有一组未使用过的键，而另一列（标识列）则必须具有从 1 开始的按顺序排列的一组号码。当数据库需要新键时，它从标识列为 1 的行取出键，然后获取一个新键以准备好供下一次使用。

更多地了解自动键生成:

要设置自动键生成，您必须对它的工作方式有所了解。要使用自动键生成，数据库必须为此设置一个表。此表具有两列:

- 增量器列存储可用的键。当数据库需要新的唯一键时，它会从此列中检索一个唯一键。
- 标识列是只有一个数字 1 的实例的数字列表。此列告知数据库要从增量器列中选择哪个键。标识列是键生成表的主键。

当数据库需要新的唯一键时，它会查找标识列值为 1 的行。此行的增量器列值保存下一个可用的键。数据库使用此键并更新表，以便下一次可以使用新键。

以下是一个键生成表的示例。此数据库的下一个可用键是什么？答案在该表后给出。

表 1. 键生成表

标识列	增量器列
3	78
4	3
1	65
2	12

此表中的下一个可用键为 65，原因是增量器列中的该键与标识列中的 1 在同一行中。

在从表中访存下一个可用键之后，就会更新表，以获得下一次需要的键。数据库还可以通过采用多行的增量器列值一次检索多个键。

简而言之，要使自动键生成工作，只需将键生成表设置为两列：一列是主键列，它用作标识列；另一列用于存储下一个可用键。必须使用这样一条记录来初始化此表：该记录的标识列值为 1，并且其增量器列值是第一个可供使用的键。一旦设置了这两列，您就可以开始使用自动键生成。

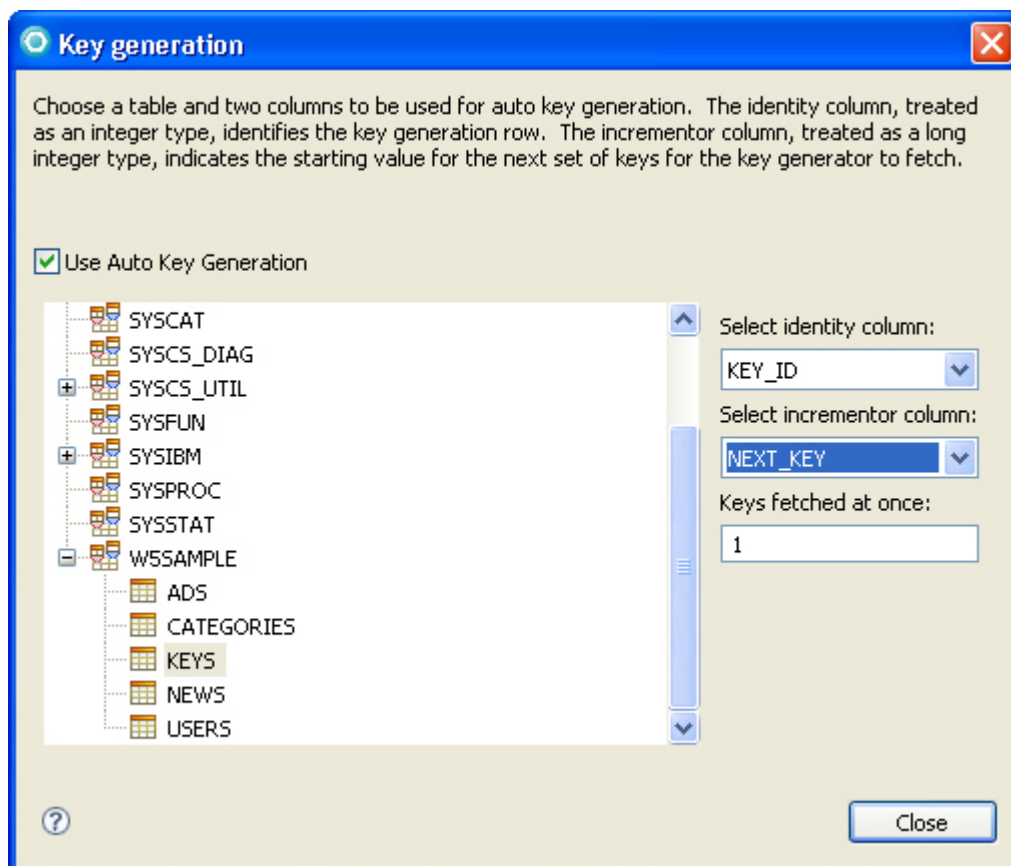
设置自动键生成

本教程附带提供的样本数据库具有一个名为 KEYS 的键生成表。如上所述，它的两个列将为每条新记录提供新的 ID 号。在这些步骤中，您将设置 create_record 关系记录以从 KEYS 表获取该记录的 ID 号。

1. 在“项目资源管理器”视图中双击 **new_record.jsp** 页。
2. 在“页数据”视图中双击 **create_record** 关系记录。将打开“配置关系记录”窗口。
3. 单击**复用现有记录或记录列表中的元数据定义和使用数据库中的现有数据来填充记录**，然后单击下一步。
4. 从下拉列表中选择数据库连接，然后单击下一步。
5. 在“配置关系记录”窗口中，位于右边的是一个链接列表。单击**自动生成键**。将打开键生成窗口。
6. 选择**使用自动键生成**。
7. 单击以展开 W5SAMPLE 列表，然后单击 **KEYS** 表。
8. 在**选择标识列**字段中单击 **KEY_ID**。
9. 在**选择增量器列**字段中，单击 **NEXT_KEY**。

由于广告标识只需要一个键，所以您将“一次访存的键数”设置保留为 1。如果需要多个键，则此设置将使数据库一次选择所有键。

“配置数据对象”窗口应类似于下图：



10. 单击**关闭**以关闭键生成窗口，然后单击**完成**以应用更改。现在，将为每条新记录自动生成 ID 字段。接下来，您必须除去 ID 输入字段，以使得用户不能输入值。
11. 通过单击 **ID:** 文本将光标放在输入表单表的顶行中。
12. 单击**表** → **删除** → **删除行**。
13. 保存该页面。

（可选）如果想要看到正在为您生成哪个键，可以不删除 ID 号的输入组件，而是将它替换为绑定至 `create_record` 的 ID 列的输出组件。这时，自动生成的键将出现在表单顶部，但是用户不能够更改该键。

测试已完成的教程

当您准备发布 Web 应用程序时，将需要一个主管该 Web 应用程序的服务器，以使用户能够通过因特网访问 Web 站点。但是，要测试该 Web 站点，可以使用一个可用的运行时来模拟服务器以进行测试。要了解如何测试 Web 站点，请参阅第 13 页的『课程 1.3: 测试 Web 站点』。

课程要点

您已经完成了课程 2.4。在本课程中，学习了如何设置自动键生成。

模块 2: 总结

您已完成“模块 2: 添加高级功能”。

已学习的课程

现在，您已了解如何完成下列任务：

- 格式化 Web 页面上的数据库记录
- 添加一个组件，它允许用户将文件从 Web 页面上载到数据库中
- 使用导航规则从一个页面自动浏览至另一个页面
- 使用自动键生成

其他资源

有关此产品或者 JavaServer Faces 和 JSF 技术的更多信息，请参阅下列链接。

相关信息



JavaServer Faces 技术



developerWorks 中的 JSF

使用 JavaServer Faces 在 Web 页面上显示动态信息

您已完成“使用 JavaServer Faces 在 Web 页面上显示动态信息”教程。本教程指导您学会了如何使用 JavaServer Faces 来创建可使用数据库中的信息的 Web 页面。

已学习的课程

通过完成本教程，您了解了下列概念和任务：

- 如何将 Web 页面连接至数据库。
- 如何创建一些页面，可以通过它们创建、读取、更新和删除数据库中的记录。
- 如何将数据从一个页面发送到另一个页面。
- 如何格式化 Web 页面上的动态内容。
- 关于 JavaServer Faces 和 JSF 组件

其他资源

如果您想要更多地了解本教程中讨论的主题，可参阅下列来源：

相关信息



JavaServer Faces 技术



developerWorks 中的 JSF