IBM

Design the layout and structure of a Web site

Contents

Design the layout and structure of a Web site 1

Design the layout and structure of a Web site

This tutorial illustrates how to design a simple dynamic Web site that links Web pages to a database. Dynamic Web sites make it possible to automate tasks on the Internet and adapt content to the needs of each person using the site.

Time Required: 60 minutes
   Related information
   View the PDF version

Introduction: Design the layout and structure of your Web site

This tutorial demonstrates how to design a simple dynamic Web site that links Web pages to a database.

Unlike a static Web site, a dynamic Web site can accept information from users, store it in a database, and change itself to display the new information. Dynamic Web sites make it possible to automate tasks on the Internet and adapt content to the needs of each person using the site.

Learning Objectives

In this tutorial, you will learn how to create a Web site that mimics the functions of the classified advertising section in a newspaper. It will display items for sale while allowing the user to add new items, change details about the items like the price, and search for a particular type of item. While this tutorial is simplified for beginners, the principles and technologies it covers are also used in larger and more complicated Web sites.

Important: Because each lesson builds on the lesson that precedes it, you should complete the lessons in numerical order.

This tutorial should take approximately 60 minutes to complete.

Skill level

Beginner

Lesson 1: Create a Web project

This lesson demonstrates how to create a Web project

The first step in working on your Web site is to create a Web project. The Web project is the container that holds all of your Web resources.

You can create a Static or a Dynamic Web project. Create a Static Web project if you plan to create a Web site that contains only simple resources, such as flat HTML files. Create a Dynamic Web project if you plan to create a Web site that contains dynamic elements, such as JavaServer Pages and servlets. In most cases, you will want to create Dynamic Web project so that you can make full use of the development tools.

In this tutorial, we will work with a dynamic Web project with JavaServer Faces support.. To create a dynamic Web project, complete the following steps:

ShowMe

1. From the Web Perspective, click **File** > **New** > **Dynamic Web Project**. The New Project window opens.
2. Type a name for the Web project in the Project name field. For this tutorial, name your project `ClassifiedsTutorial`.
3. On the facets page of the wizard, select the following facets: Base Faces Support, JSTL, and Design-Time Page Template support.

The wizard creates a new project with the name and options you selected

Lesson 2: Design the structure of the Web site

This lesson demonstrates how to design the structure of the Web site.

Web pages contain content, such as HTML elements, images, and links; however, a Web site is more than just a collection of Web pages. A Web site is the hierarchical design and organization of how the Web pages fit together. A Web site should have a high-level goal (such as providing an organized collection area for classified listings), and each Web page should serve a specific purpose in meeting that goal (for example, a page that searches through the listings).

When designing a Web site, you should consider what functions the site should support to achieve the overall goal. For a Classified Web site, there are several functions that you want the site to perform. For example, you probably want site visitors to be able to search the listings as well as create and edit new listings.

To make a Web site that performs these functions, we will plan four pages in this exercise:
• A home page that shows all of the classified ads on the database.
• A page that adds a new classified ad to the database.
• A page that changes or deletes an ad currently on the database.
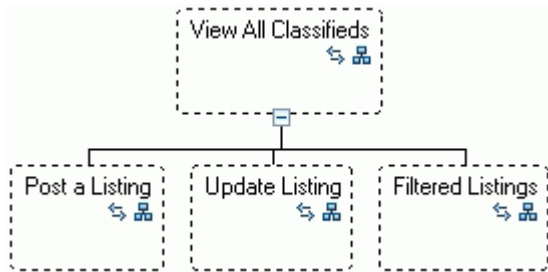• A page that shows search results by filtering the ads by category.

We will create the pages themselves in the next exercise.

ShowMe
1. In the Project Explorer view, expand ClassifiedsTutorial.
2. Double-click **Web Site Navigation** in the Project Explorer view. Web Site Designer opens to the Navigation view. With this tool, you can add, delete, and rearrange the Web pages that compose the Web site.
3. At first, there are no page icons displayed in the Navigation view. Typically, the first page that you consider when designing your Web site is the top page, or the home page of the site. This is the page where visitors enter your Web site. Drag the **New Page** component from the Palette view and drop it onto the Navigation view. A page icon displays with the navigation title of the page in focus.
4. This page will be the home page that displays all of the ads on the database. Type `View All Classifieds` for the navigation title of the page. Later you will learn how the navigation title is used as the page label for site navigation purposes.
5. For site visitors to post their own listings, they need a page for creating new listings. Add a second page icon by right clicking on the **View All Classifieds** page icon and selecting **Add** → **New Page** → **As Child**
6. Name the new page `Post a Listing`. Notice that the new page icon is displayed beneath the **View All Classifieds** page icon and connected with a line. This means that Post a Listing and View All Classifieds have a parent-child relationship.
7. You also need a page where visitors can edit or delete existing listings. For example, if a visitor has added a listing for a bicycle and it has been posted for a long period of time with no response, the

seller might want to lower the asking price. Add another page icon as a child of **View All Classifides** and name it `Update Listing`. Notice that this new page icon is a sibling of Post a Listing, and another child of View All Classifieds.

8. Next, your Web site needs a search function to enable visitors to search for existing listings. To add a page for displaying search results, create another page icon as a child of **View All Classifieds** and name it `Filtered Listings`. **Note:** If you accidentally place a page in the wrong place, or if you want to reorder your page structure, you can drag the page icons at any time to rearrange your site.

9. Press **Ctrl+S** to save your site design. The Navigation view should look like this:

Notice that there is a row of small icons across the bottom of each page icon. The first icon, which is not yet visible, represents actual files associated with the icon and will appear after those files are created. If you create a JSP file, the icon is a blue diamond (◆); if you create an HTML file, the icon is a set of brackets (◇). The second (⇆) and third icons (品) are visible, indicating that by default, all of the pages are set to appear in navigation and in site maps respectively. The fourth icon (⬚), which is not yet visible, diplays when you have associated a page icon with an actual file that file does not contain any Web site navigation. The icon (⬤), which is not yet visible at the top of the page icon represents the navigation root. You will learn more about these icons and the other facets of the page icon display as you work through this module.

Now that you have created your general Web site structure, you can start the next lesson by clicking on **Next.**

Lesson 3: Populate the Web site with Web pages

This lesson illustrates how to populate the Web site with Web pages

Once you have your Web site laid out using page icons in Web Site Designer, you are ready to begin creating your pages. The page icons are currently bordered by broken lines; this indicates that the file represented by the page icon, or the actual page, does not exist. After the files are created, the page icons will have solid lines as borders

ShowMe

1. For this tutorial, you need to have the Base Faces Support facet installed for your project. If you did not select this facet when you created the project, you can do so now by selecting **Project** → **Properties** → **Project Facets**. This facet enables you to create JSP Web pages that are enabled to use JavaServer Faces technology.

2. Right click the page icon for the View All Classifieds page, and from the pop-up menu click **Create Page**. The New Web Page wizard appears.
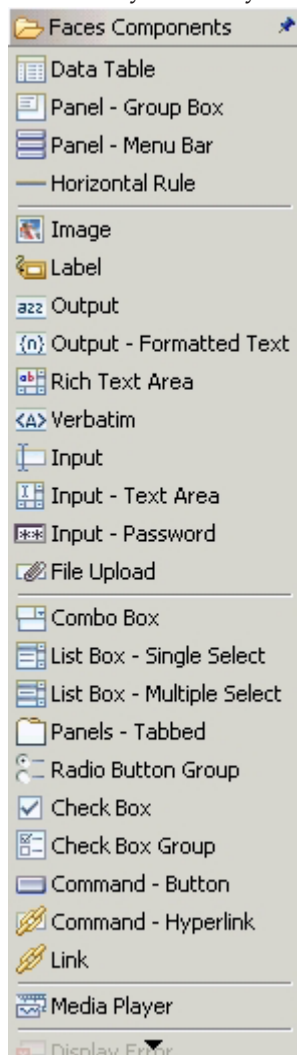
Learn more about JavaServer Faces and Faces components:

JavaServer Faces is a technology developed by Sun Microsystems that helps you build user interfaces for dynamic Web applications that run on a server. The JavaServer Faces framework manages UI state across server requests and offers a simple model for the development of server-side events that are

activated by the client. JavaServer Faces is based on a model-view-controller (MVC) framework. For JavaServer Faces, this means that the controller is a servlet, the model is represented by JavaBeans™, and the view is comprised of JavaServer Faces components with little or no application code. The goal of this model is to separate content from presentation.

In Rational® Software Development Platform, tools such as Faces components/are available to help you use this technology in your Web applications. There is a JavaServer Page (JSP) custom tag library for expressing a JavaServer Faces interface within a page that the Rational Software Development Platform has conveniently used to create a Faces JSP wizard. This wizard helps you create JSP files that are enabled to use the Faces components. Faces components let you develop a Web application by dragging components from a Faces drawer in the Palette view and dropping them on the Web pages you are creating. For example, you can quickly drag an input text field and drop it to a form on the page. Then you can drag a Submit button next to the input text field. Finally, you can connect the input text field to your data source. This would enable end users to enter data from the Web application to your data source.
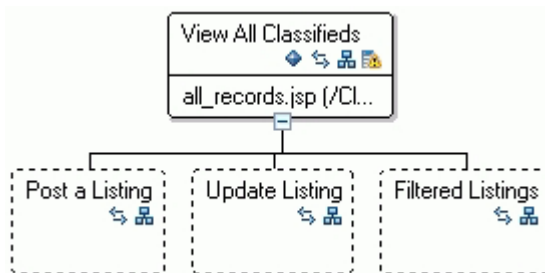
Another benefit of applications built using Faces components is that the user interface is rendered independently of the underlying program logic. This means you can visually create an application that can dynamically access data and run equally well on a browser or hand-held device.

You can also learn how to use Faces components to add data connections to your Web site in the Display dynamic information on Web pages with JavaServer Faces tutorial.
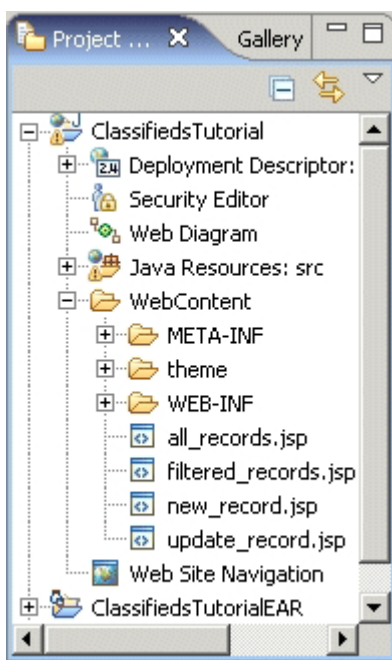
3. Accept the default folder (/ClassifiedsTutorial/WebContent).

4. In the **File Name** field, type the name for your new file, `all_records`. Though it can be convenient to name your file the same as the navigation label you assigned the page when laying out your site, you can also name your files in a way that makes sense for the resources on your local system, and use the navigation labels only as a naming convention for your future navigation.

5. Ensure that **HTML** is selected in the **Markup Language** field, which you can view by selecting the **Options** button. For the rest of the fields, accept the defaults.

6. Click **Finish**. The new file is opened in Page Designer. Save and close the file. The Web Site Designer appears again.

7. You can now see that the View All Classifieds icon has a solid border and a small blue diamond icon () to indicate that a JSP file has been associated with it, while the other page icons still have a broken border to show that files have not yet been created for them.

8. Continue creating JSP pages for the other page icons in the same manner. For this tutorial, follow this naming convention:

   • Post a Listing - new_record
   • Update Listing - update_record
   • Filtered Listings - filtered_records

9. Press **Ctrl+S** to save the Web Site Navigation.

When you are finished creating the four pages, your Project Navigator view shows all of the new resources under the WebContent folder:

You have now created blank pages for each of the pages you will have in your Web site. To start the next lesson, click **Next**

Lesson 4: Design the look and feel of the Web site

This lesson demonstrates how to design the look and feel of the Web site using a page template.

As a unit, the Web pages that comprise a Web site should have a similar visual design and layout. The individual pages should appear related, so that the Web site is cohesive. It is good practice to begin a project by thinking first of the Web site design and then creating the individual Web pages. Otherwise, you could get a Web site that consists of several disjointed Web pages that have no organization or apparent visual relationships.

The best way to create a common look and feel for your Web site is by using a page template. A page template is a single file that you can use to control common elements on all of the pages in your site. After you apply the template to the pages, you can minimize the effort of future design changes by only making the change once on the template; the change is then applied to all of the pages that use the template. You can create two kinds of Web page templates: design-time templates and dynamic templates. **Dynamic page templates** use Struts-Tiles technology to generate pages on the web server at runtime. Pages that you create using **Design-time templates** are created immediately. This tutorial demonstrates the use of design-time templates.

The page template is designed to have common areas and content areas. A *common area* is a shared area common to all pages that use the page template, which is useful for elements that are the same for every page, such as site banners and navigation areas. The *content area* will be different for each page. After the template is applied to individual pages, you can edit the content areas in those pages to add information specific to that page. Page elements such as text and images specific to a particular page are the types of elements that go into content areas of the page template. You can create as many content areas and common areas as you want in a template.

A page template controls the look and feel of a site layout in a very different way than using a style sheet.

Learn more about the difference between page templates and style sheets:

Page templates and style sheets control the common look and feel of a Web site in very different ways, though they can be used to complement each other. For example, you can use a page template to control the site's layout and content areas, and use a style sheet to control the visual display of tags, such as font size and color. In general, page templates control the location of data on the page, while style sheets control the appearance of that data.

Page Templates

A page template is a model for other pages. To make a template, start by creating an ordinary Web page and adding elements to it, such as pictures, text, and tables. Every element placed on the template will appear on the individual pages linked to the template. This ensures that the pages will look the same. Then, add content areas to the template. The content areas are the parts of the template where each individual page will have its own content. When you open a page that is linked to a template, you can edit only inside the content areas of the template.

Style Sheets

Style sheets control the appearance of text, tables, images, and other elements on Web pages. For example, you can set all of the first-level headings on your site to be the same color, size, and font. When you change the definition for first-level heading text in the style sheet, the matching text on every page changes to match it.

For this tutorial, we are going to create a page template and define some content and common areas in order to design a consistent look and feel for the entire Classifieds site. This page template ensures each page looks like the following design:

Creating a new page template

1. In the Project Explorer view, right-click your project name, ClassifiedsTutorial.
2. Select **New** › **Web Page Template** from the pop-up menu. The New Web Page Template file wizard opens.
3. Accept the default folder (/ClassifiedsTutorial/WebContent). This is where your template file will be located after it is created.
4. Type a name for your template file in the File Name field. For this tutorial, name the file `template`. Note that the resulting file's full name will be template.jtpl.
5. Select the **Design-time Template** radio button.
6. Ensure **HTML** is selected as the markup language. You can see this and other information by selecting the **Options** button.
7. Click **Finish**. The new template is opened in Page Designer. When a blank template is opened, you are reminded in a dialog box that you must add at least one content area to the template. Click **OK** to close the dialog box.

Adding some elements to the blank template

Now that the template has been created, you need to add some elements to the blank template.

1. Begin by deleting the text `Place content here`.
2. One of the easiest ways to lay out the structure of a page template is by using an invisible table. This way, you can specify sections of the page to be used in different ways, and you can control the placement of objects on the page.
   a. In the Palette view, expand the **HTML Tags** drawer by clicking on it.
   b. Drag the **Table** component from the palette onto your blank template. The Insert Table dialog opens.
   c. For this template, you will need 3 rows and 1 column. Type 3 in the **Rows** field and 1 in the **Columns** field.
   d. Then, click **OK**. The table displays on your template page.
3. At first, the table will appear very small. You need to edit the attributes of the table to adjust the size and structure of the table.
   a. Right-click the table and select **Properties** to open the Properties view.
   b. In the Properties view, click the **Table** tab.

c. In the **Alignment** field, select **Center**. This centers the table on the page.

d. You need to increase the overall size of the table so that the structure of the template will expand depending on the size of the screen that the page is displayed on. Otherwise, the design of your pages will appear at a fixed size and could appear very large on small screens, or very small on large screens. In the **Table width** and **Table height** fields, enter 90 and select the percentage (%) sign.

e. Because you do not want the outline of the table to actually appear on your pages, you need to also set the **Border** to 0 pixels. Page Designer displays invisible table borders as dotted lines.

4. Because you want a banner at the top of every page announcing the Web site name and visuals to tie the site together, you need to format the area for it in this layout table. Also, though you likely would have a graphic designer create a logo or banner image for your site, you can add text as a placeholder for the banner in this template.

a. Select the top cell in the table.

b. With this cell selected, specify a **Height** of 70 pixels on the **TD** tab in the Properties view. If you knew that you were creating a template that needed to accommodate a banner graphic that your graphic designer was creating, you could specify that this cell be the exact size of the banner graphic.

c. In the Properties view on the **TD** tab, also set the **Color** to **Gray** in the pull-down menu. If you wanted to use a different color, you could use the eyedropper tool to pick a color from anywhere on the screen, or you could type the RGB value (such as #808080 for gray) into the field.

d. Click anywhere in the top row and type Welcome to the Classifieds!

e. Because the default text is small and hard to see, you need to increase the size of the font. Select the whole phrase and click **Format** > **Font**.

f. To give the site a newspaper-feel, select **Courier** as the **Font**. Select **6** as the **Size**, and select **White** as the **Color** to stand out against the Gray background of that cell.

g. Then, click **OK**.

h. To center the banner text, select the text again and click **Format** > **Align** > **Horizontal Center**.

5. To allow users to go to different pages, you need a row of links beneath the banner. You will add tabs for navigation later, but for now, format the second row to leave room for the buttons.

a. Select the second cell in the table.

b. Set this cell to the same size as the banner cell by entering a **Height** of 70 pixels in the Properties view on the **TD** tab.

6. The third row is where your content will be located. You need to ensure that the content will align with the top of the row.

a. Select the third row in the table.

b. In the Properties view on the **TD** tab, select **Top** for the **Vertical Alignment**.
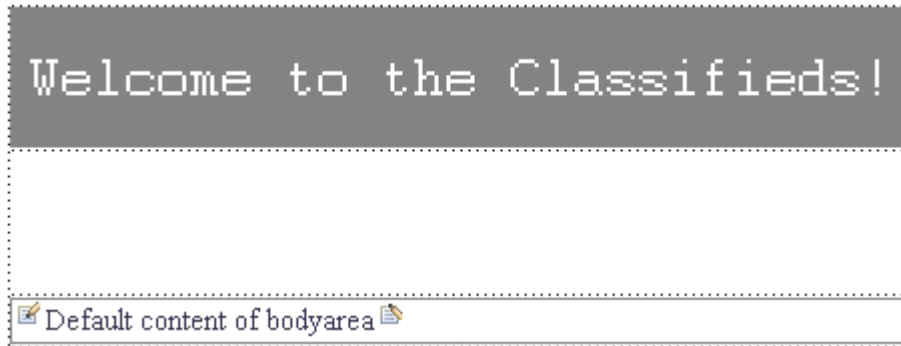
Adding a content area

ou need an area of the template where the specifics of each page can be displayed, such as the search results on the filtered_records page, or the form to fill in a new posting on the new_record page. Now that you have the common structure and color scheme defined for the site, you are ready to add a content area.

1. On the Palette view, click the **Page Template** drawer.

2. Drag a **Content Area** component into the third row. This is where your individual pages will supply specific content. The Insert Content Area for Page Template wizard opens.

3. You can accept the default **Content area name** of **bodyarea** by clicking **OK**. The content area name allows you to specify different names for different content areas for organization purposes. For instance, if you were designing a site that would always have two content areas on every page, you could name the area that will be populated with the main content **bodyarea** and the area that will be populated with search results **searcharea**. This also helps if you need to apply your page template to

pages that already exist, because you can then assign areas of existing pages by markup tags to populate specific content areas of the template. The content area is inserted into the table cell.

4. Save your changes to the page template.

Your basic template structure should now look like this sample:

You have now created a page template with some common and content areas defined for your Web site. Click **Next** to begin the next lesson.

Lesson 5: Apply a page template to the Web site

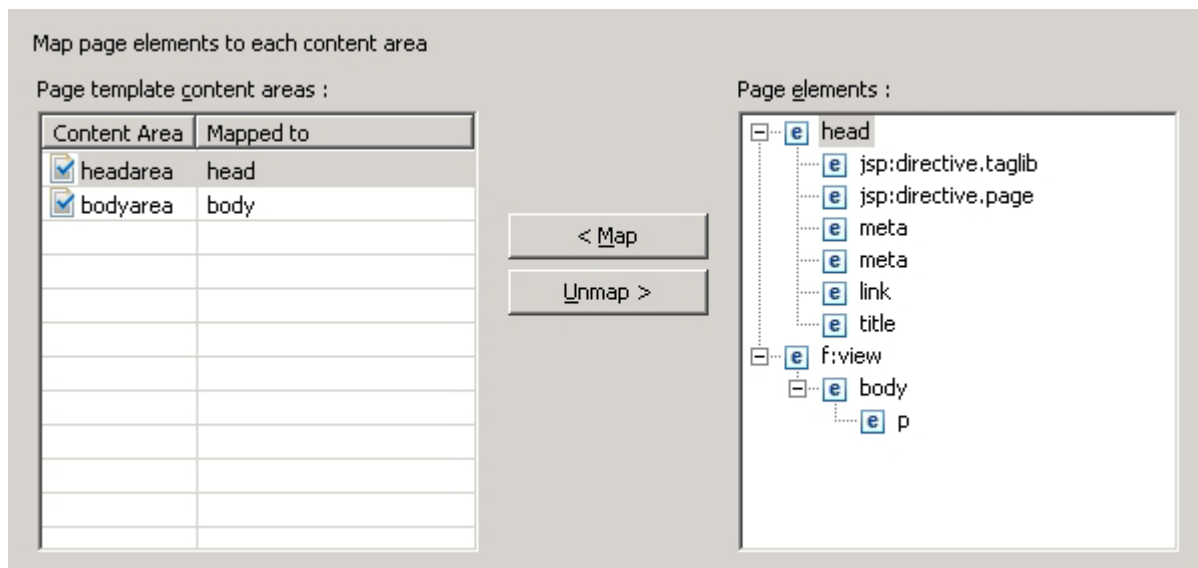This lesson illustrates how to apply a page template to the Web site

Now that you have made some preliminary design decisions about how your Web pages will be laid out using your page template, you need to apply the template to the pages in your site. Once you apply the template to the individual pages, every change in the template will appear in each Web page.

1. Return to your site's structure by double-clicking **Web Site Navigation** in the Project Explorer view.
2. Select all four page icons by clicking and dragging the pointer to create a selection box enclosing the entire area around the page icons. All of the icons within the box will be selected.
3. Right-click and select **Set Template** > **Apply Template**. The Apply Page Template wizard opens.
4. Expand the **My Templates** folder in the Templates box. If you did not create your own template, you could use one of the sample page templates provided.
5. Select your template and click **Next**.
6. Because the pages in the Web site do not contain any content yet, it is not important how we map page elements to the template content areas, so you can accept the defaults and click **Next**. If you already have content in the pages you wanted to apply the template to, you can select a certain page as an example page for mapping existing elements to template content areas.

   Learn more about mapping content areas:

   You can apply page templates to Web pages that already contain some content and still be sure that the page will look the way you expect once the template elements have been added. If you have Web pages that already contain content, you can map certain sections of the existing page to content areas defined in the page template.

   When you apply a page template, there is a page in the wizard called **Map page elements to the page** template:

The source for the content areas is defined in the **Page template content areas** list. The elements for the Web page that you selected to apply the template to display in the **Page elements** list. In the Page template content areas list, you can select a defined content area, and in the Page elements list, select the tag element to which you want to map the content area. Then click **Map**. Do this for each of the content areas. This associates the content areas in the page template with elements in your Web page.

For example, if you select one of the sample page templates, a **headarea** and a **bodyarea** content area are predefined. You can select the headarea content area and associate it with the HEAD (anything between the start and end HEAD tags) in your existing file. Any attributes defined in the page template for the headarea content area are applied to the HEAD region of the existing Web page. The preview boxes display the current page and a preview of what the page will look like once the template is applied, so you can see what the existing page will look like once the template elements are applied. You can use these preview boxes to help you decide how to map the template's content areas. All content areas must be associated with regions before you can finish applying the page template. **Tip:** By selecting a page element in the Page elements list, you can highlight a specific element in the preview of the current page. Also, you can view an enlarged preview by right-clicking **Preview of current page** and selecting **Show original size** from the context menu **Note:** Though you have only created one content area named **bodyarea**, you may notice a second content area named **headarea**; **headarea** is included in page templates by default.
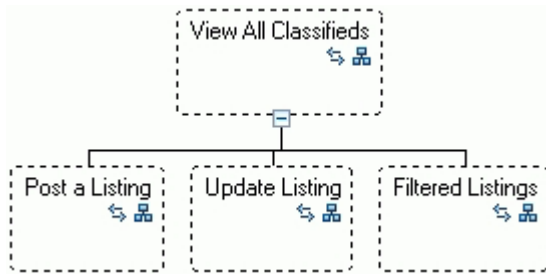
7. Again, you are able to accept the defaults for mapping by clicking **Next**. These options control how any existing content in the pages is placed inside the content areas of the template.

8. Ensure all four of your pages appear in the **Pages to apply page template** box, and that all four are selected. Note that on this page of the wizard, you can validate the application of the template to ensure that using the template will not introduce any markup errors in the pages.

9. Click Finish. The template is applied to the pages, and the page icons are updated with blue bands to reflect that each page has a template associated with it.

10. Save the Web Site Navigation.

Click **Next** to begin the next lesson.

Lesson 6: Set up a Web site's dynamic navigation

This lesson illustrates how to set up the Web site's dynamic navigation.

When you designed the structure of your Web site in Lesson 2, you made some decisions about the relationships between the pages. Recall the way you laid out the page icons with lines connecting them:

There are several different ways that one page can be related to another. These relationships are referred to like the relationships in a family. In this example, View All Classifieds is the parent page and the other three pages are its child pages. Also, these three pages are sibling pages to each other.

These page relationships are used to generate navigation links like navigation bars and tabs. Each page automatically creates its navigation links based on the pages to which it is related. This process is called *dynamic navigation*.

Checking site structure

You want to ensure that your site structure has been set up to specify which pages should be included in navigation. You can specify in Web Site Designer which pages should be included as links in dynamic navigation, which pages should be included in site maps, as well as some other navigation options that are based on the structure of your site and your page relationships. This is useful when you may have pages that need to be included in the site, but would not make sense as part of your navigation features. For example, you might have an error page that appears when the site encounters a problem; however, you would not want to show the error page as part of the navigation.

1. Double-click **Web Site Navigation** in the Project Explorer view to view your site structure.
2. You can see that all of your pages have the **Show in Navigation** option selected by default because the **Show in Navigation** icon () is visible. This option must be selected for each page that you want to appear in a navigation bar. You can also disable a page, so it will not appear in the navigation. You need to disable the search results page, Filtered Listings, from the navigation because the user will be sent there from the search option on the View All Classifieds page.
   a. Right-click the **Filtered Listings** page icon.
   b. Select **Set Navigation** → **Show in Navigation** to uncheck it. The **Show in Navigation** icon should now be unavailable on the **Filtered Listings** page icon.
3. You also need to disable your Update Listing page because the user will be sent there from the update option on the View All Classifieds page. Right-click the **Update Listing** page icon. Next select **Set Navigation** → **Show in Navigation** to uncheck it. The **Show in Navigation** icon should now be unavailable on the **Update Listing** page icon.
4. You need to set a Navigation Root for your site. The top page, usually the first page created in the site, is often set as the Navigation Root. Right-click the **View All Classifieds** page icon and click **Set Navigation** → **Set Navigation Root**. The icon () at the top of the page icon is the **Root** icon. The navigation root is important because it determines the link levels used when generating Web site navigation, such as navigation bars. **Note:** The default options set by the site navigation for the View all Classifieds page and the Post a Listing page should remain selected.
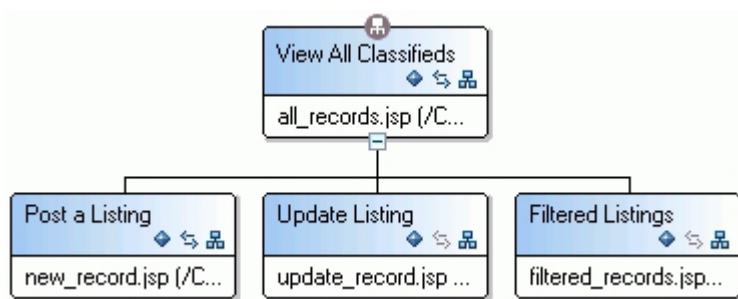5. Press **Ctrl+S** to save your changes.

Adding a navigation component to your page template
1. Return to your page template by double-clicking template.jtpl in the Project Explorer view.
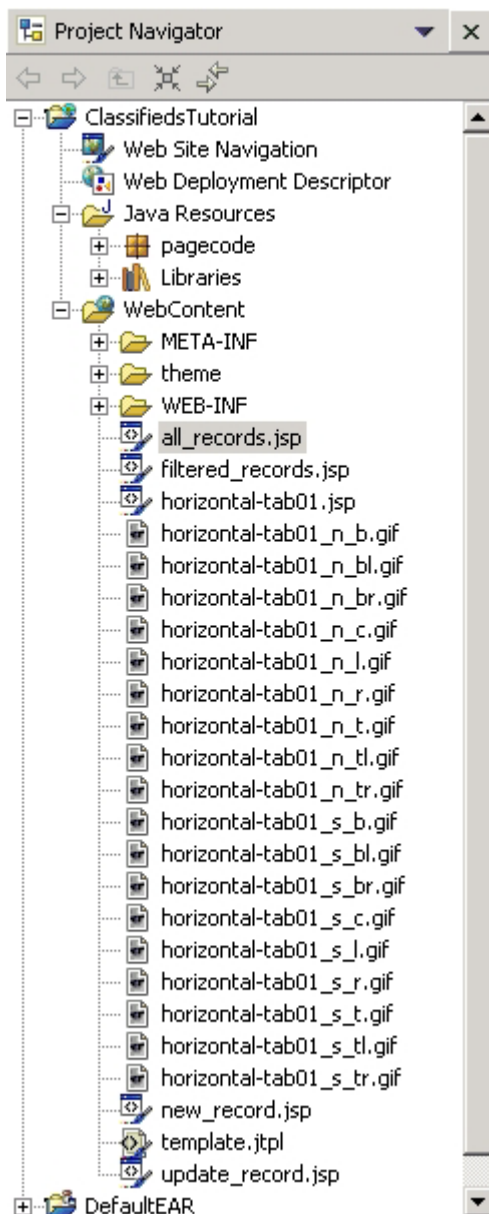
2. In the Palette view, click **Web Site Navigation** to open the drawer that contains the navigation components. There are many different types of dynamic navigation components that you can add to your Web site, including a site map. The most basic component is the Horizontal Bar, which comprises of a row of links based on the site structure.

3. Though you know you need a row of links at the top of each page allowing users to access the other pages in the site, you also want to add some visual interest to the navigation. Drag the **Horizontal Tabs** component to the second row in your page template; this component creates the dynamic links in a tabbed row. The **Insert Horizontal Tabs** wizard opens.

4. Accept the default to use a **Sample** navigation component.

5. Accept the default in the **Thumbnail** images (horizontal-tab01.jsp) and notice that the **File name** field is automatically populated. Click **Next**.

6. Select the type of links that you want to appear in the navigation bar, based on page relationships. To specify that the top page of your site, and all of each page's siblings and children will appear in the navigation, ensure that the following pages are selected:
   - Top page
   - Children of top page
   - Sibling pages
   - Current page

   Notice that if you select previous or next links, you can supply a label to appear for those links, such as Back and Forward.

7. Click **Next**. Accept the defaults on the Optional Settings for Specification File page of the wizard. If you want to generate the links at runtime using JSP custom tags, you would select **JSP Navigation**. If you want to generate the links at design time using HTML custom tags, select **HTML Navigation**. If you are adding an HTML custom tag in a Faces JSP page click the **Use Faces link** check box to create an outputLink tag for Faces.

8. Click **Finish**. The **Navigation Component** icon (◁▥▷) displays on the page template.

9. Save your changes to the page template. When you save the page template, the dynamic navigation changes are applied to all of the pages in your site.

10. Double-click **Web Site Navigation** in the Project Explorer view to view your site structure. Because your Web pages now have a navigation component, the icon (🖾) indicating that the page does not have a Web site navigation does not appear on the page icon.

The navigation component adds several files to your Web project. After saving your template, notice that your Project Explorer view contains a new .jsp and several .jpg files. These files display the tabs in your pages:

Project Navigator

- ClassifiedsTutorial
  - Web Site Navigation
  - Web Deployment Descriptor
  - Java Resources
    - pagecode
    - Libraries
  - WebContent
    - META-INF
    - theme
    - WEB-INF
    - all_records.jsp
    - filtered_records.jsp
    - horizontal-tab01.jsp
    - horizontal-tab01_n_b.gif
    - horizontal-tab01_n_bl.gif
    - horizontal-tab01_n_br.gif
    - horizontal-tab01_n_c.gif
    - horizontal-tab01_n_l.gif
    - horizontal-tab01_n_r.gif
    - horizontal-tab01_n_t.gif
    - horizontal-tab01_n_tl.gif
    - horizontal-tab01_n_tr.gif
    - horizontal-tab01_s_b.gif
    - horizontal-tab01_s_bl.gif
    - horizontal-tab01_s_br.gif
    - horizontal-tab01_s_c.gif
    - horizontal-tab01_s_l.gif
    - horizontal-tab01_s_r.gif
    - horizontal-tab01_s_t.gif
    - horizontal-tab01_s_tl.gif
    - horizontal-tab01_s_tr.gif
    - new_record.jsp
    - template.jtpl
    - update_record.jsp
  - DefaultEAR

Previewing your navigation

If you want to see an instance of how the navigation looks on the pages rather than in the template, double-click all_records.jsp in the Project Explorer view and then click the Preview tab in Page Designer. You can see that the names that appear on the tabs are the navigation labels for the pages, rather than the file names. Also, notice that the current page is displayed as the highlighted tab; as you navigate through the site, the current page will always be the highlighted tab on top.

Tip: If you change the structure of your site in the Navigation view, the dynamic navigation links will automatically update to reflect the new structure.

Welcome to the Classifieds!

| 🖊 View All Classifieds^abc | 🖊 Post a Listing^abc |

Place content here.

Now that you have set up the dynamic navigation, continue to the next lesson by clicking on **Next**.

Lesson 7: Preview the Web site

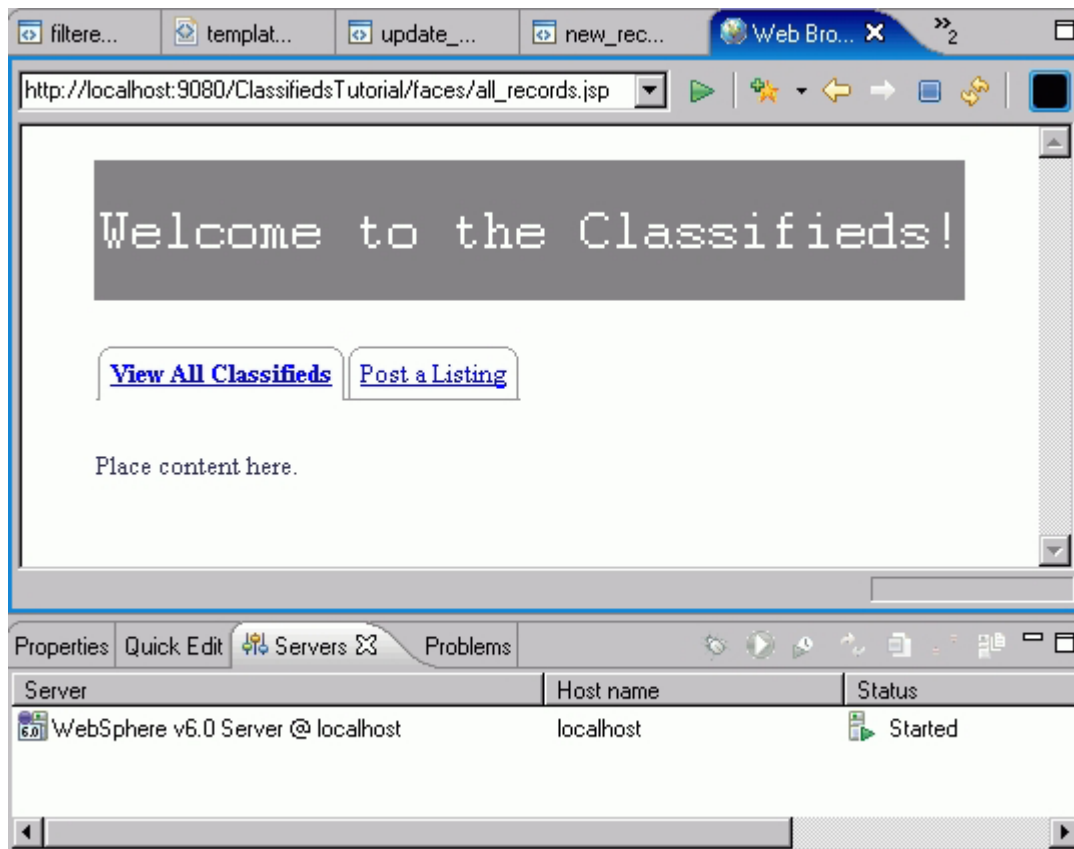This lesson illustrates how to preview the Web site on a server.

ShowMe

When you are ready to publish your Web application, you need a server that will host the Web application so that users can access your Web site through the Internet. However, to test your Web site, you can use the default test environment included with Rational Software Development Platform called the WebSphere® test environment. When you use the WebSphere test environment, the server tools will simulate a server for you.

Note: At any time during your Web site development, you can open a page in Page Designer and use the **Preview** tab to see how your design will look in a browser; however, the Preview view does not allow you to see the dynamic aspects of your page (such as database connections) as they would display running from a server.

1. Right-click the **all_records.jsp** file in the Project Explorer view.
2. Select **Run As** → **Run on Server**. The Run on Server wizard appears.
3. If this is your first time testing pages, you will need to create a new testing server. Select **Manually define a server**. If you have already created a server, select **Choose an existing server**, select your server, and click **Finish.**
4. Ensure **WebSphere v6.0 Server** is selected in the **Select the server type** list.
5. Click **Finish.**

The server tools create the new server simulation, start it, and open the page in a simulated Web browser. This may take a moment. In the Servers view, you can watch the messages as the server tools start the server, as shown here:

You can view the complete server startup log at *install_dir/runtimes/base_v6/profiles/default/logs/server1*, where *install_dir* is the directory where the product is installed. You can now click through the skeleton of your site to see how the links work, and how the site will display once it is published to a Web server

Now that you completed the tutorial, you can view the tutorial summary by clicking on **Next**.

Summary: Design the layout and structure of your Web site

Congratulations! You have completed the tutorial Design the layout and structure of your Web site.

Lessons learned

If you have completed all of the Lessons, you should now understand:
- How a Web project contains the resources for a Web application, what a Web site is, and how it is comprised of Web pages.
- The importance of creating a cohesive Web site rather than starting with page design.
- The concepts of a site structure (including family and peer hierarchies), dynamic navigation and relational linking.
- The benefits of using a page template and content areas to control the look and feel of a site.

Additional resources

If you want to learn more about the topics covered in this tutorial, consider the following sources:
- Web site development in the online help.
- The tutorial **Display dynamic information on Web pages with JavaServer Faces**.
- The tutorial **Import, test, and modify a Web site**.