

Oracle® Portal

Building Advanced Portals

Release 3.0

November, 2000

Part No. A86183_02

ORACLE®

Oracle Portal Building Advanced Portals Release 3.0

Part No. A86183_02

Copyright © 1996, 2000, Oracle Corporation. All rights reserved.

Primary Author: Vanessa Wang

Contributing Authors: Marcie Caccamo, Frank Rovitto, and Susan Barton

Contributors: Tim Harkness, Jeanne Kramer, William Lankenau, Steven Leung, Dave Mathews, Darren McBurney, Dmitry Nonkin, Nancy Teh, Todd Vender, Harry Wong, Hui Zeng

The Programs (which include both the software and documentation) contain proprietary information of Oracle Corporation; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. Oracle Corporation does not warrant that this document is error free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of Oracle Corporation.

If the Programs are delivered to the U.S. Government or anyone licensing or using the programs on behalf of the U.S. Government, the following notice is applicable:

Restricted Rights Notice Programs delivered subject to the DOD FAR Supplement are "commercial computer software" and use, duplication, and disclosure of the Programs, including documentation, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement. Otherwise, Programs delivered subject to the Federal Acquisition Regulations are "restricted computer software" and use, duplication, and disclosure of the Programs shall be subject to the restrictions in FAR 52.227-19, Commercial Computer Software - Restricted Rights (June, 1987). Oracle Corporation, 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy, and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and Oracle Corporation disclaims liability for any damages caused by such use of the Programs.

Oracle is a registered trademark, and PL/SQL is a trademark or registered trademark of Oracle Corporation. Other names may be trademarks of their respective owners.

Contents

Send Us Your Comments	xi
------------------------------------	-----------

Preface.....	xiii
---------------------	-------------

1 What Is Oracle Portal?

1.1	How Does Oracle Portal Work?	1-1
1.1.1	Pages	1-3
1.1.1.1	What are "portlets"?	1-3
1.1.2	Content Areas.....	1-5
1.1.2.1	What's the Difference Between Content Areas and Pages?	1-6
1.1.3	Applications	1-6
1.1.4	Security	1-7
1.1.5	Putting It All Together	1-7
1.2	Examples in the Building Advanced Portals manual	1-8
1.2.1	Simple intranet portal.....	1-9
1.2.2	Line of business portal	1-9
1.2.3	Human Resources portal	1-9
1.2.4	Information repository	1-9

2 Planning Your Web Content

2.1	Planning Pages	2-1
2.1.1	Step 1: Determine Your Audience	2-1
2.1.2	Step 2: Get The Data	2-2
2.1.3	Step 3: Develop a Look and Feel	2-4

2.1.4	Step 5: Organize the Page	2-4
2.1.5	Step 6: Consider Security	2-5
2.2	Planning Content Areas	2-6
2.2.1	Gathering Information	2-7
2.2.2	Information Analysis	2-12
2.2.3	Troubleshooting Your Content Area Structure	2-24
2.2.3.1	All of My Content Areas Have the Same Categories	2-24
2.2.3.2	Item Level Security is Required Everywhere	2-24
2.2.3.3	I Do Not See Links to My Perspectives	2-24
2.2.3.4	My Categories Do Not Appear in the Content Area Map	2-25

3 Creating an intranet portal and adding simple portlets

3.1	Create a page	3-3
3.2	Add customized HTML code as a portlet	3-4
3.3	Add and customize a Favorites portlet	3-5
3.3.1	Add the Favorites portlet	3-5
3.3.2	Add a new favorite link	3-6
3.3.3	Remove an existing favorite link	3-7
3.4	Create an event calendar and add it as a portlet	3-7
3.4.1	Create an event calendar using the Create Calendar Wizard	3-8
3.4.2	Create a form to update the calendar	3-10
3.4.3	Change the appearance of the form	3-12
3.4.4	Create a database sequence for the form	3-14
3.4.5	Create a link between the form and the calendar	3-16
3.4.6	Create a button on the calendar to access the form	3-17
3.4.7	Add the calendar as a portlet	3-18
3.5	Displaying the portal to others	3-19
3.6	Summary	3-19

4 Creating a line of business page for your intranet portal

4.1	Create a tabbed page and add a page as a portlet	4-3
4.2	Add an existing folder as a portlet	4-5
4.3	Create a sales chart portlet that links to a report	4-6
4.3.1	Create the report	4-6
4.3.2	Testing the sales report	4-9

4.3.3	Create a link to your sales report	4-10
4.3.4	Create a sales chart.....	4-11
4.3.5	Test the chart with links	4-12
4.3.6	Display the chart as a portlet.....	4-12
4.4	Create a multi-tabbed region containing two charts.....	4-14
4.4.1	Use the available charts.....	4-14
4.4.2	Add the charts to the same portlet.....	4-15
4.5	Create a multi-tabbed region containing various components	4-16
4.5.1	Create a sales commission chart	4-17
4.5.2	Create a sales commission report.....	4-18
4.5.3	Create four LOVs for the commission search form.....	4-21
4.5.3.1	Create JOB_LIST dynamic list of values	4-21
4.5.3.2	Create REGION_LOV dynamic list of values	4-22
4.5.3.3	Create DEPARTMENT_LOV static list of values	4-22
4.5.3.4	Create EMP_LOV dynamic list of values with a bind variable	4-23
4.5.4	Create a commission search form	4-24
4.5.5	Add the chart, report, and form to the same portlet	4-27
4.6	Create a customer entry form.....	4-28
4.6.1	Create a database sequence	4-28
4.6.2	Create a customer entry form.....	4-30
4.7	Create a product report.....	4-32
4.8	Create a menu containing links	4-35
4.8.1	Create two links for your menu.....	4-35
4.8.2	Create a menu.....	4-36
4.8.3	Add the menu as a portlet	4-38
4.9	Summary	4-39

5 Creating a Human Resources portal with security

5.1	Create three tabs for your intranet portal	5-5
5.2	Create an employee search form that links to an organization chart	5-8
5.2.1	Create a team organization chart.....	5-8
5.2.1.1	Create a company-wide organization chart	5-11
5.2.2	Create an employee search form that links to the team organization chart	5-12
5.3	Create components to display and post job vacancies	5-15
5.3.1	Create a form to look up a specific job vacancy.....	5-15

5.3.2	Create a form to post or close a job vacancy.....	5-17
5.3.2.1	Create a database sequence for the form.....	5-17
5.3.2.2	Create three LOVs	5-18
5.3.2.2.1	Create the Manager dynamic list of values.....	5-18
5.3.2.2.2	Create the HR_CONTACTLOV dynamic list of values	5-18
5.3.2.2.3	Create the Vacancy_Status static list of values	5-19
5.3.2.3	Create the form to post or close job vacancies.....	5-19
5.3.3	Create a link to connect the vacancy report with the lookup form	5-23
5.3.4	Create a report that displays all current job vacancies	5-24
5.4	Create a vacation request and approval system.....	5-26
5.4.1	Create a vacation request form.....	5-26
5.4.1.1	Create a database sequence for the form.....	5-26
5.4.1.2	Create EMP_LIST LOV	5-26
5.4.1.3	Create the Vacation Request form	5-27
5.4.2	Create a vacation approval form	5-29
5.4.3	Create a calendar that displays requested vacation dates	5-31
5.4.4	Create a vacation balance report	5-32
5.5	Implement security to control display of components	5-34
5.5.0.1	Apply security to the Vacation Calendar	5-35
5.5.0.2	Apply security to the Vacation Approval Form.....	5-36
5.5.0.3	Apply security to the Vacation Balance Report.....	5-36
5.5.0.4	Apply security to the Advertise Vacancy form.....	5-37
5.6	Design the tabs of the HR portal	5-37
5.6.1	Add portlets to the Human Resources tab.....	5-37
5.6.2	Build the Managers tab	5-39
5.6.3	Build the HR Representatives tab	5-40
5.7	Summary.....	5-42

6 Creating an information repository

6.1	Upload content and folder structure as a zip file.....	6-2
6.2	Create categories	6-5
6.2.1	Assign a category to an item	6-6
6.3	Create perspectives	6-7
6.3.1	Associate an icon with a perspective.....	6-8
6.3.2	Assign a perspective to an item.....	6-9

6.4	Create custom types	6-10
6.4.1	Create attributes	6-10
6.4.2	Create a custom item type.....	6-12
6.4.3	Add a course description.....	6-13
6.4.4	Create a custom item type with a procedure call	6-14
6.4.5	Add a benefits definition.....	6-16
6.5	Change the layout of a folder	6-17
6.5.1	Change region properties	6-18
6.5.2	Add an item to a region.....	6-21
6.5.3	Move an item to a different region.....	6-22
6.6	Change folder access privileges	6-23
6.7	Publish objects as portlets	6-25
6.8	Summary	6-25

A Oracle Portal Administrator Tasks

A.1	Creating a user who can build these examples.....	A-1
A.1.1	Create a user.....	A-1
A.1.2	Grant the user access to the casebook application	A-2
A.1.3	Enable the user to publish objects as portlets	A-2
A.2	Creating a group	A-3
A.3	Installing the demo portals, schema, and content areas	A-4
A.4	Grant access privileges to the schema.....	A-4
A.5	Create three groups for the Human Resources portal	A-4
A.5.1	Create three users for the groups.....	A-4
A.5.2	Create the HR Managers and HR Representatives groups	A-5
A.6	Create an Inside Healthy Living Repository	A-6

Glossary

Index

Send Us Your Comments

Oracle Portal Building Advanced Portals, Release 3.0

Part No. A86183_02

Oracle Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for revision.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, then indicate the document title and part number, and the chapter, section, and page number (if available). You can send comments to us in the following ways:

Electronic mail: webdbdoc_us@oracle.com

If you have problems with the software, contact your local Oracle Support Services.

Preface

Welcome to the Oracle Portal Building Advanced Portals Manual! The examples in this book are intended to help you learn about Oracle Portal and its advanced features.

Intended Audience

This book is intended for anyone who uses Oracle Portal to build portals. The needs of advanced users are addressed. If you are new to Oracle Portal and portal technology, refer to the *Oracle Portal 3.0 Tutorial* contained on the Oracle Portal 3.0 CD.

Each chapter contains step-by-step instructions to build a portal and its related applications from start to finish. Each portal that you build will demonstrate how to use many of Oracle Portal's powerful features.

Prerequisites

This book is intended for advanced Portal users, who are familiar with the Oracle Portal Release 3.0 environment. If you are unfamiliar with Oracle Portal, or would like to learn more about the basic concepts of Oracle Portal, see the *Oracle Portal Tutorial*.

You must have a Web browser installed on your machine. The minimum and recommended requirements are:

- Microsoft Internet Explorer 4.0.1 (Service Pack 1) or 5.0
- Netscape Communicator 4.0.8 or 4.72

You should be familiar with using your Web browser. For example, you should know how to start your Web browser and be familiar with the basic user interface elements, such as hypertext links.

You must set up or obtain the following information from your portal administrator:

- A user name and password to log on to Oracle Portal
- The URL of the Oracle Portal home page
- The sample portals and content areas provided on <http://technet.oracle.com>. Contact your portal administrator, or refer to Appendix A.3, "Installing the demo portals, schema, and content areas" for more information.

For information about installing and configuring Oracle Portal, see the *Oracle Portal Release 3.0 Configuration Guide*.

Example Scenario

For the purposes of the examples contained in this manual, you are building portals for employees of Healthy Living Wholesalers, a company that provides health and fitness products.

Structure

This book contains the following chapters:

Chapter 1, "What Is Oracle Portal?"	Briefly introduces you to the Oracle Portal environment and concepts.
Chapter 2, "Planning Your Web Content"	Provides recommendations for building a portal, and requirements you should keep in mind before beginning development.
Chapter 3, "Creating an intranet portal and adding simple portlets"	Explains how to build a page and add portlets. Also describes how to build a calendar application using links and database sequences.

Chapter 4, "Creating a line of business page for your intranet portal"	Explains how to build a portal specifically for a line of business: the Healthy Living Wholesalers Sales department. Describes how to add pages to an existing portal, as well as how to create report, form, menu, and chart components. Also demonstrates how to create multi-tabbed portlets, dynamic and static lists of values, and database links.
Chapter 5, "Creating a Human Resources portal with security"	Explains how to build a portal for a Human Resources department, where three different types of users have customized views of the portal. Describes how to implement security to pages and tabs, as well as individual components. Also demonstrates how to build hierarchies and reports based on SQL queries. Shows how to leverage information from an existing content area in the portal.
Chapter 6, "Creating an information repository"	Explains how to build a content area to contain information for a company, such as HTML pages, text files, and images.
Appendix A, "Oracle Portal Administrator Tasks"	Describes how to create a user for Oracle Portal who can create portals in the database.
"Glossary"	Explains terminology specific to Oracle Portal.

Related Documents

For more information, see the following manuals in the Oracle Portal Release 3.0 documentation set :

- *Oracle Portal Release 3.0 Tutorial*
- *Oracle Portal 3.0 Release Notes*

Conventions

The following conventions are also used in this manual:

Convention	Meaning
bold text	<p>Bold text indicates a term defined in the text, the glossary, or in both locations.</p> <p>Bold text also indicates button names, labels, links, and other user interface elements.</p>
<code>monospace text</code>	Monospace text indicates text that you need to enter.
<i>italic text</i>	Italic text is used for emphasis, and to indicate a cross-reference to another manual.
<>	Angle brackets enclose user-supplied names. Substitute an appropriate value.
[]	Brackets enclose optional clauses from which you can choose one or none.

What Is Oracle Portal?

This chapter introduces some basic information about Oracle Portal, and provides a brief introduction to the examples presented in this book. Sections included are:

- Section 1.1, "How Does Oracle Portal Work?"
- Section 1.2, "Examples in the Building Advanced Portals manual"

1.1 How Does Oracle Portal Work?

Oracle Portal helps you efficiently manage, access, and interact with information by enabling you to create three distinct entities: **content areas**, which provide a built-in structure for organizing, classifying, and cross-referencing items in a Web site; **applications**, which allow you to insert, maintain, and display data from your Oracle 8i database; and **pages**, which make data from your content areas, applications—even sources outside your intranet--accessible from a single location. Even after you become familiar with these concepts, using the product to its fullest potential requires a solid understanding of how all the pieces fit together. Let's take a look at the following illustration:

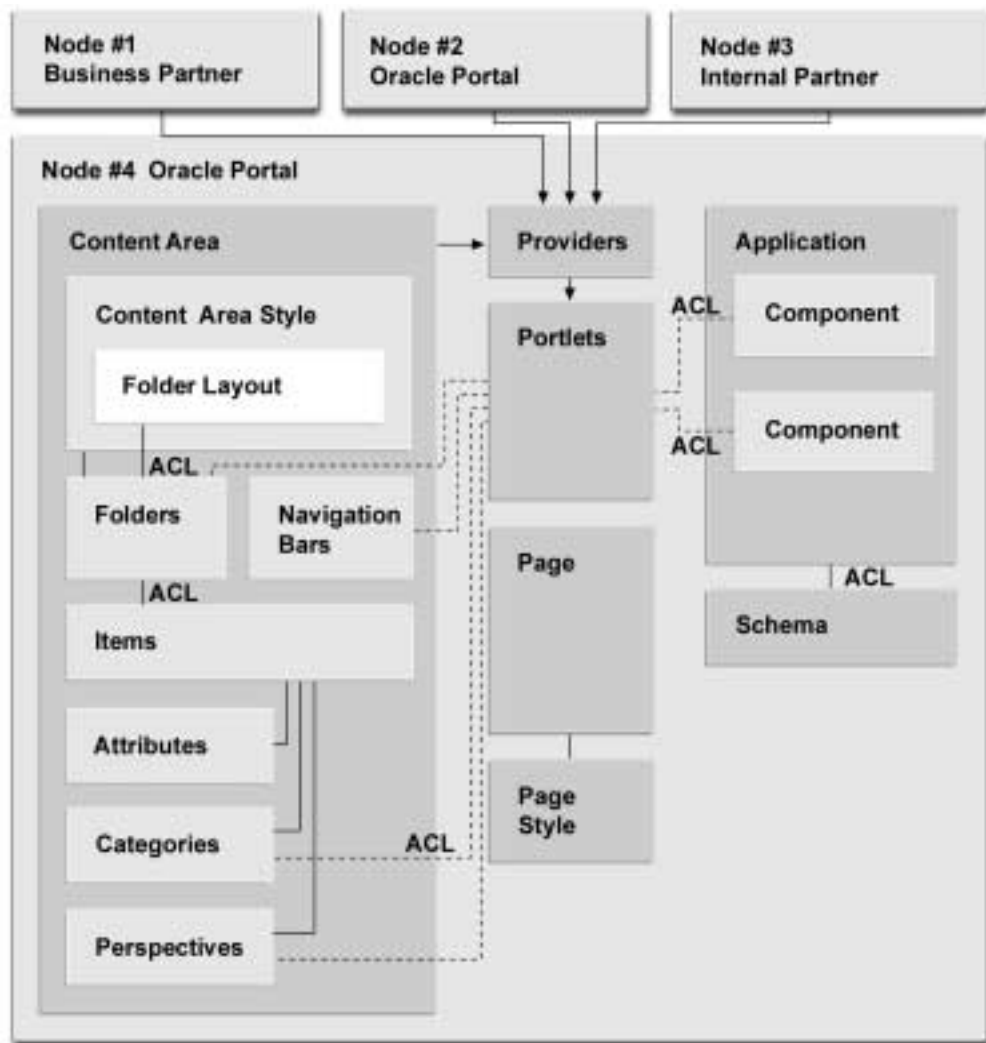


Figure 1–1 Oracle Portal Content Areas, Pages, and Applications

It is important to realize that the placement of pages, portlets, and portlet providers in the center is significant. Although you can use Oracle Portal to create content areas and/or applications without ever creating or using pages, the focus of Oracle

Portal is really to enable you to build portals. A **portal** is a Web-based application that enables your users to access content areas, external Web sites, other applications, newsfeeds, and other useful information. In very simple terms, a portal makes chunks of information, usually from disparate data sources, accessible from a single entry point. That entry point is called a **page**. For example, a financial analyst's page would likely include relevant financial information from real-time Internet-based stock quotes, financial reports from an online repository, and access to the legacy financial accounting and banking systems. None of these data sources are inherently similar, or even know anything about each other; yet with Oracle Portal, they can all be presented side-by-side, right on the same page.

1.1.1 Pages

As mentioned, a page is really the face of the portal: what users see and use to interact with the content of the portal. The appearance of the page--that is, the colors and fonts in which the page is rendered--is controlled by the **page style**.

Each page is divided into rectangles or squares called **regions**. Within each region, you can place one or more **portlets**. A portlet is a re-usable information component that summarizes or provides access to an information source. Portlets are the fundamental building blocks of an Oracle Portal page.

1.1.1.1 What are "portlets"?

You can think of portlets as "Web components" which display excerpts of other Web sites and generate summaries of key information. These portlets can then be collected with other portlets so that users have easy access to frequently used sites and information, all from one page. Oracle Portal uses portlets to integrate both structured and unstructured data into a single, personalized view.

You can use portlets to access nearly any type of Web-accessible information, such as files published on the corporate intranet, reports on data managed by enterprise applications, and news or stock quotes from the Internet.

Here are some common uses for portlets:

- *Centralizing access to intranet sites.* Within large organizations with many intranet sites, users are often confused about where to go to find the specific information they need to do their jobs. Oracle Portal makes it easy to gather links to all these sites in one place, then organize these access points so that users can readily access frequently-used sites from their own personalized portal.
- *Publishing information and documents.* You can easily put content on the Web using Oracle Portal because all editing and file uploading can be done through

your Web browser. You can arrange your content into folders, then place these folders as portlets into any portal. Doing so enables you to customize the specific content you see on that page.

- *Integrating dynamic data services.* You can use portlets to render content provided from external data sources in your own portal. For example, you can display real-time news stories and weather reports.
- *Providing an interface to Web application.* Organizations may have many different applications which users need to perform their duties, but it is not always efficient for users to frequently navigate through entire applications to find a small piece of specific, dynamic information. Instead, you can use portlets to automatically log in to a particular application (or its data source), retrieve a summary of the necessary information, and display the summary as a part of the portal.
- *Integrating with other corporate systems.* Organizations have many different, and perhaps incompatible, systems within their intranets for performing different functions. Portlets enable you to present a consistent interface to the various systems.

A portlet belongs to a *portlet provider* that has been registered with Oracle Portal. Each portlet can have one and only one provider; a provider may have one or more than one portlet that exposes an underlying application or data source. Applications, content areas—even other pages—can all be portlet providers, because elements from each can be placed within a region on a page as portlets. Within a content area, navigation bars, folders, categories, and perspectives can all be published as portlets. Within an application, components such as menus, forms, reports, and so on can become portlets. The dashed lines in Figure 1–1 represent the ability of each object to be published as a portlet.

For example, you might create a navigation bar in a content area, publish the navigation bar to the portal, then include the navigation bar as a portlet on a page. If Yahoo! was a portlet provider, you might have portlets such as local television listings, your calendar, your Yahoo email account, and so on. Oracle Portal provides instructions for creating your portlet providers in the Portal Development Kit (PDK), so you can even make data from your proprietary applications available to your pages. For more information, see *Oracle Portal Release 3.0 Development Kit* or visit <http://technet.oracle.com>.

Each provider is defined, registered, and maintained within a single instance (*node*) of Oracle Portal. Once a provider is registered with an Oracle Portal node, its portlets are available to be placed on a page. A **local node** is the instance of Oracle Portal to which you are logged on; a **remote node** is another instance. In Figure 1–1,

Node 4 is considered the local node. Although Node 4 is the only node shown in detail, Node 3, which contains another instance of Oracle Portal, would most likely embody a similar structure. Nodes 1 and 2 represent participants in the Oracle E-Business Portals Partner Initiative, which is dedicated to fostering a community of outside partners to develop content and solutions built around Oracle Portal. The solid lines tipped with arrows from each of these nodes suggest that they have been added to the local node as remote nodes. By doing so, the local node gains access to all portlets available on these nodes for use on locally created pages.

Oracle Portal provides two methods of creating portlets. You can use the Oracle Portal user interface to either leverage external information sources or to create your own applications or content areas that display as portlets.

1.1.2 Content Areas

Note: For the purposes of this chapter, a sample content area focusing on the entertainment industry has been created. See Table 2-1 on page 6 and Table 2-2 on page 7 for a complete description of the Entertainment content area.

Each content area is made up of **folders**, in which **items** are stored. Within the Entertainment content area, for example, there is a folder for Dance, one for Music, and so on. Within each folder are items: files, URLs, images, even other folders that relate to the folder in which they are contained. An item in this content area might be a review of an event written up by a reporter from the local newspaper.

Just as the appearance of a page is controlled by the **page style** applied to the page, so is the appearance of a folder controlled by the **content area style** applied to the folder. **Navigation bars**, which help you move quickly to key areas of the content area, are also controlled by content area styles. Content area styles and page styles are distinct entities, however: you cannot apply a content area style to a page, nor a page style to a folder or navigation bar. The layout of a folder is also determined by the content area style.

Following along with the diagram, next come some classifications that help characterize an item:

- An **attribute** helps describe the item. For example, every item has attributes for the display name, description, and when the item should appear on the content area.
- Each item must be assigned to a **category**, which helps answer the question “What type of item is this?” In the Entertainment content area, the categories Artist/Performer, Poster, Review, and so on have been established. Users can search on a category to gather information from all of the folders. By searching

on the category Recording, for example, all recordings from Dance, Music, Movies, and the other folders are returned.

- Finally, an item may be further characterized by any **perspectives** assigned to the item. Perspectives are intended to answer the question, “Who might be interested in this item?” The entertainment perspectives include time period (17th Century to 1990s), audience (Adult, Family, Teens), venue (Live or Outdoors), and content (Image, Audio, etc.), among others.

For information on how to plan your content areas, see “Planning Content Areas” on page 6.

1.1.2.1 What’s the Difference Between Content Areas and Pages?

Now that you’ve reviewed the basic concepts behind content areas and pages, you may still be wondering how they really differ, and how you can determine which one(s) you need to build. There are two key differences:

- Create a content area when you have many users who need to store and manage a lot of information in a hierarchical structure. Create a page when you want to present each person or group with a different collection of information from a variety of sources (which may or may not include data from a content area).
- While a content area often makes sense to the people maintaining it, it can sometimes be difficult for consumers to find the information they need. A page, on the other hand, exposes *just the information the user needs*, regardless of where it exists in the content area’s folder structure.

For example, suppose the folder structure in your content area reflects how responsibilities are distributed within your organization, rather than embracing a more logical structure easily understood by everyone. People who have been in the organization for a long time may know to look under the Human Resources folder for information on Facilities, since they report through the same management chain, but users outside your organization or those new to the group won’t be privy to this insider information. However, by publishing the Facilities folder as a portlet and placing it on a page, users who need this information can access it quickly, without having to know or care where it resides in your Web site.

1.1.3 Applications

Oracle Portal **components**, such as menus, reports, forms, calendars, and so on, may be used individually or may be connected to produce complete Web-based

database **applications**. For example, an employee expense chart may link to a report that details individual expenses. When you publish an application's component as a portlet, that application becomes a portlet provider.

Each application you build in Oracle Portal is based on a *schema* in the database. In addition, you can create and manage other database objects for use by application components, such as schemas, tables, views, stored procedures, and so on. For example, you might create a table object and use it to manage data inserted/updated/deleted by a form component. The data within the table can then be displayed to users through a chart or report component.

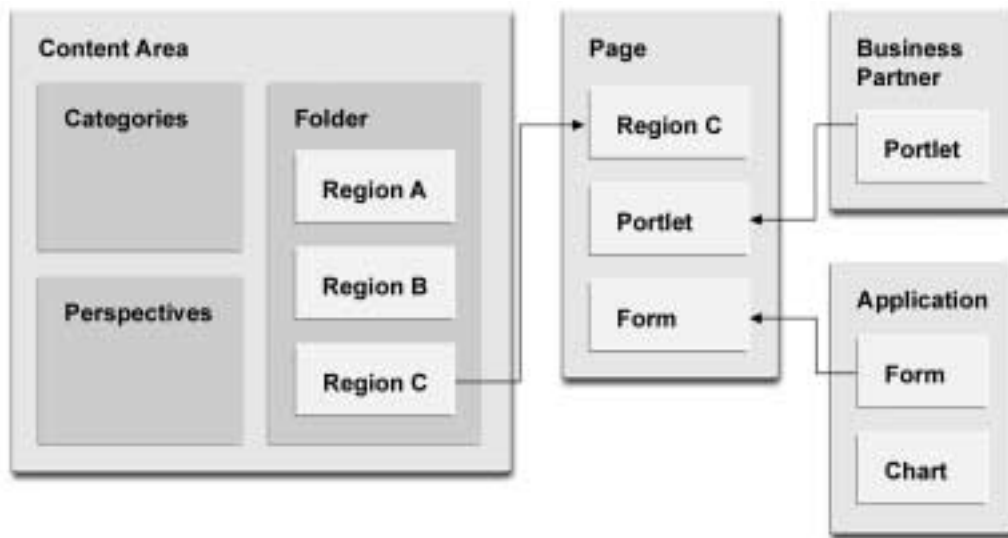
1.1.4 Security

The letters ACL you see next to some of the boxes stand for **Access Control List**. The ACL controls which users and groups may access the object, and to what extent. For example, if you wanted all the users in a group to be able to see the items in a folder on a content area, in the folder's ACL you would grant that group View privileges. Content areas, folders, and items all have ACLs, as do pages, components, schemas, and applications.

ACLs are just one way you can control access to objects within Oracle Portal. You can also use global privileges to grant access to all objects of a given type. For example, granting the Create privilege for All Pages to a group enables all members of the group to create pages. (If the ACL and a global privilege conflict, the higher privilege level prevails.)

1.1.5 Putting It All Together

Now let's see how all these pieces might look on a page. The following illustration depicts a very simple example of how you might bring together data from three different sources together on a single page:



Notice that the folder in the content area contains regions, as do pages. A **folder region** contains items; a page region contains portlets. As you can see, only one of the folder regions depicted actually appears on the page. You can do this by publishing the folder as a portlet, then using the Customize link to exclude all other folder regions except the one you want to display.

Another region on the page contains a form from an application you built with Oracle Portal. Finally, you populate your third page region with a portlet provided by an external business partner. This simple example demonstrates how easy it is to create completely customized sets of information for individual users or groups.

Note: For a graphical introduction to Oracle Portal, visit the Oracle Portal Quick Tour in the online help system or at <http://technet.us.oracle.com>.

1.2 Examples in the Building Advanced Portals manual

The examples contained in this manual are intended to provide you with a starting point for building your own intranet portals using Oracle Portal. This manual describes two scenarios: the basic intranet portal and a line of business portal.

1.2.1 Simple intranet portal

This portal example teaches you how to build a basic page for your intranet, and add pre-built portlets that are provided with Oracle Portal. You will build a customizable HTML portlet, where you can add your own HTML. You will also learn how to build a calendar application, where users can add, delete, and modify events. In this example, you'll learn how to create a form, a calendar, and a link.

1.2.2 Line of business portal

This portal example teaches you how to add a page to the simple intranet portal. You will learn how to build chart components, then build a tabbed portlet that accesses these charts. You will also build several forms, reports, and folders and expose them as portlets on your portal page. You will also learn how to build a menu portlet to organize existing documents and data, as well as documents you create.

1.2.3 Human Resources portal

This portal example teaches you how to control the display of components and tabs within a portal to enable you to create a customized workplace. You will learn how to build a team organization chart, as well as reports based on SQL queries. you will also learn how to build complex applications to display different information on the portal, depending on the type of user. You will also learn how to leverage information in an existing content area in a portal.

1.2.4 Information repository

This content area example teaches you how to build a repository of information that contains all types of information, including text, documents, and images. You will learn how to use a content area to create an information repository that provides Healthy Living employees with access to all the information they need in their day-to-day lives within the company.

Planning Your Web Content

This chapter explains the planning steps you should take before you start using Oracle Portal. In particular, it covers a set of techniques you can use to define the overall structure of your Web content. Defining this structure greatly simplifies the process of actually creating content areas and pages.

Note: A section that addresses the planning considerations for applications will be added in a future release.

This chapter contains the following sections:

- Section 2.1, "Planning Pages"
- Section 2.2, "Planning Content Areas"

2.1 Planning Pages

This section briefly discusses some of the issues you should consider during each phase of the planning process. Of course, planning and creating pages successfully is largely a matter of experience: the more you create pages and use them, or receive feedback from others who use them, the better you'll be able to determine what works and what doesn't.

You'll find this section particularly helpful if you've never created Oracle Portal pages before.

2.1.1 Step 1: Determine Your Audience

The first thing you need to think about is who will use the page(s) you are about to create. Are you creating a single page that only you will see, or are you responsible for creating separate home pages for several different groups or individuals? If you're creating your own page, of course, you have considerably more freedom, as you can experiment with Oracle Portal to see what you like and don't like. If you're

creating several different pages, however, you should spend some time thinking about what kind of information each user or group requires.

Ask yourself the following:

- Is there some information that everyone requires, such as a corporate bulletin board?
- Are there tools or applications that certain groups depend upon and would like to access quickly? It's probably wise to meet with the user or group who will use your page to ensure you have accurately identified their needs.
- Have the groups and users already been defined by the portal administrator? If not, you may want to see that this happens fairly quickly, as you'll need the user and group accounts to establish proper security.
- Are you creating a content area to contain the data at the same time you're creating pages to access that data? This scenario involves many considerations:
 - The content area and pages may well be used by different audiences. Make sure the folder structure reflects the needs of your audience, as described in Fill out the Item Worksheet.
 - Be sure to publish each folder as a portlet if you're planning to expose it on a page. Since the page merely exposes the folder (or a portion of it), all updates to the folder are automatically reflected on the page.
 - Keep in mind that content area attributes, such as categories and perspectives, can also be published to the portal and placed on pages for the appropriate audiences.
 - If you want to publish only certain items to the portal, keep those items contained within a single folder region. When you display a folder as a portlet, you can select which regions within the folder you want to expose.

Hint: Take a sheet of paper, write the name of each group or user across the top, and list the required information/applications for each.

2.1.2 Step 2: Get The Data

After you've listed the information required by each user or group, your next step is to determine how to make that information available as portlets so you can plug them into the appropriate pages.

Note: A lot of what you can do here depends upon how familiar you are with the content areas and applications established for your enterprise. You may need to enlist the aid of the individual or a group member for whom you are creating the

page to find out the names of content areas, content area objects, applications, and/or components.

- Take a look at the first group or user. Is any of the data required from an Oracle Portal content area or application? If so, check the portlet repository to see if the data and/or applications have already been published as portlets.

Order the repository by Provider and see if any of the content areas or applications you need are listed as providers. If you see a portlet that corresponds to a need you identified in step 1, place a check mark next to it on your worksheet. Keep in mind that content areas and applications appear as providers *only* when an object belonging to that content area (like a navigation bar or folder) or application (like a form or menu) has been published as a portlet. If you don't see a content area or application listed, but you know the object you're looking for exists, use the Navigator to edit that object and make sure the **Publish As Portlet** check box has been selected.

After you've reviewed the portlet repository by Provider, you might want to change the View By criteria to Category to see if you've overlooked any information that might be useful for a particular user or group. Of course, this functionality is useful only if your portal administrator has already classified your portlets by category.

- Is any of the information you need available on another node? If you don't know the answer to this question, see if you can obtain a user name and password to other instances of Oracle Portal so you can review those portlet repositories, or talk to someone familiar with those nodes. If you identify a portlet on another node you'd like to access, ask your portal administrator to add that node as a remote node so that its portlets and providers appear in your portlet repository.
- Are any of the required applications external to your enterprise? If so, ask your portal administrator to register them as External Applications. Then you can add the External Application portlet to your page, customized to include only the applications required by the user/group. Not only does this provide quick access to the application, but if the user registers his or her user ID and password, Oracle Portal will log on to the application for the user behind the scenes each time it is launched.
- Do you have any static information you want to present, such as a company announcement or instructions for completing a task related to a particular group? If so, you can use the HTML portlet provided by Oracle Portal, which allows you to enter HTML code, text, and hyperlinks to create any content you wish. Note that there is a browser limit of 30,000 bytes of text.

If the data or application you need does not exist, but should, you'll probably have to contact the appropriate person to either add the necessary content to a content area or create the required application or component.

2.1.3 Step 3: Develop a Look and Feel

Once you've obtained the data you need, it's time to start thinking about the look and feel of your page. The appearance of your page is primarily controlled by the page style, although the page banner has a big impact as well. When you're creating your page style, you should scroll through the list of Available Style Elements and make a conscious decision about each.

You may want to work with a graphic designer when doing so, and when considering these issues:

- Does your organization or company have a color scheme you'd like the page to reflect? If you can, obtain the RGB values for these colors so your page will be an exact match.
- Do you already have a .jpg or .gif file you can use as your page banner, perhaps containing the name of your organization? If not, is a corporate logo available to include in your banner?
- What color or image do you want to use as a page background? If you choose an image and a color, and the image is small enough, the image is repeated as a pattern on top of the page color. If the image is too large, it overlays the page color completely.
- If maintaining visual consistency is not important to you or your organization, consider selecting <Use User's Default Style> for the page. This allows each user to view the page in the color and text settings he or she prefers.
- If you're planning to add tabs to your page, remember that you can design custom images and use those for the tabs instead of the default. If you do so, however, you will not be able to change the tab name, color, and so forth using Oracle Portal, as those elements are part of the image itself. You can also use images for active and inactive tabs.

2.1.4 Step 5: Organize the Page

Organizing the page means determining how to arrange the portlets you want to include so that they are both visually appealing and useful. You may want to sketch out some alternatives on a piece of paper before you start working in the product itself.

To organize your page, you'll need to know the following:

- Roughly how many portlets will your page contain?
- Does your page require a navigation bar? If so, where should it be placed?
- Do you have enough information so that tabs might be helpful? Keep in mind that the more tabs you add, the longer it takes your page to render. Also, avoid nesting tabs within tabs, which not only degrades performance but also overwhelms users.
- Is there is a portlet you want to appear on every tab--say, one containing links to your corporate Web page? If so, divide your page into two regions: one containing your tabs as shown above, and the other containing the common portlet. Only the region containing the tabs can be subdivided into regions unique to each tab.

2.1.5 Step 6: Consider Security

The last step in the planning phase is to think about who will need access to this page, and to what extent.

- Should the page be available to everyone, even public users who do not log on? If so, select **Display Page To Public Users** on Create Page's Access tab.
- Do one or more tabs need a different level of protection from the page on which they appear, or can they inherit the page's ACL? It may make sense to add a tab to a page that is otherwise uncustomizable on which users are free to add whatever they please.
- If you've decided to restrict access to a group, should everyone in that group see the same page? If so, grant the View privilege to the group. This means that users will be unable to add any extra portlets to their page, or hide or show the portlets you've added. In addition, everyone will see the same Recent Objects and Favorites portlets--they won't be customizable by each individual. If you don't want to impose these restrictions, assign either Full Customization or Add Only Customization privileges to the group.
- Besides the Access tab, there is another way to control access to pages: through user and group global privileges. A global privilege grants access to a user or group for all objects of a particular type. For example, if a group has the Manage privilege for pages, every user in that group has full authority over every page created on that instance of Oracle Portal. Global privileges allow you to grant many users access to a page simultaneously, rather than adding each user individually through the Access tab.

2.2 Planning Content Areas

In a classic Web site, the smallest component is a page. On the page are an assortment of links, images, and text. In Oracle Portal content areas, the smallest component is an *item*, displayed within a container called a *folder*. Folders are very much like folders in a file system, with multiple items existing within a folder. The term *page* in Oracle Portal is reserved for an entity on which information is displayed in the form of *portlets*, which may or may not include data from your content area. For more information on planning pages, see Section 2.1, "Planning Pages".

Each Oracle Portal item has a set of *attributes*. These attributes include information such as title, description, and author. Each item must also be assigned to a *category*: for example, General or Presentation. Users can display category pages to access all of the items in the specified category quickly. *Perspectives* provide another dimension of classification. Each item may be assigned to many different perspectives. Users may also display perspective pages to quickly access all of the items in the specified perspective.

For convenience, the same sample content area, based on an entertainment concept, is used throughout this chapter. The content area's folder, category, and perspective structure is described below.

Table 2–1 Entertainment Content Area Folder Structure

Folder	Subfolders
Dance	Ballet, Ballroom, Jazz, Modern, Swing, Tribal
Music	Alternative, Blues, Country, Folk, Jazz, Pop, Rock, Rhythm & Blues
Movies	Action/Adventure, Comedy, Drama, Horror, Mystery, Romance
Theater	Comedy, Drama
Visual Art	Two Dimensional, Three Dimensional

Table 2–2 Entertainment Content Area Categories and Perspectives

Categories	Artist/Performer Exhibit, General Historical Reference Performance Poster Recording Reproduction Review Venue Web Site/Index
Perspectives	17th Century 18th Century 19th Century 20th Century 1950s 1960s 1970s 1980s 1990s Live Outdoors Adult Content Family Friendly Age 0-6 Age 6-12 Teens Ticket Information Image Content Audio Content Video Content Full Color Black and White

Notice that one of the categories is called General. This category is created by Oracle Portal and, because it is the default, cannot be deleted. You must always have at least one category in your content area. Notice also that the 20th Century and Family Friendly perspectives have several child perspectives each. Both categories and perspectives may have child categories and perspectives

2.2.1 Gathering Information

The key to planning your content area is understanding the kind of information that will be stored in it and how users will most often use the information. To make this

process easier, we recommend that you use the worksheets provided in this chapter to help you gather the necessary facts to design a successful content area.

Step 1 Get Organized

To prepare for the information gathering phase of designing your Oracle Portal content area, make copies of the following worksheets.

Figure 2–1 Item Worksheet

Item Description:

Who should be allowed to add this content?

Who should be allowed to view this content?

Who should control who can add/view this content?

Possible Classifications:

Mark the core content for this item:

File_____ Text Item_____ URL_____ Folder Link_____

PL/SQL Call_____ App. Component_____ Image Map_____ Other:_____

Standard Attributes for Items:

Name, Title, Description, Category, Perspectives (multiple), Author, Expiration Date, Image, Rollover Image, Keywords

Additional Attributes (may be displayed with item and used for searching):

Figure 2–2 User Community Worksheet

Owners	Content Providers	Viewers	Folders	Sample Items

Figure 2–2 User Community Worksheet

Owners	Content Providers	Viewers	Folders	Sample Items

Step 2 Fill out the Item Worksheet

Complete a copy of the item worksheet for an item that you believe belongs in your content area. Try to come up with as many different example items as you can--don't bother to create multiple examples of the same kind of item. At this point, you should not worry about folders, categories, and perspectives. You will consider them in a later step.

Here is a sample of a completed item worksheet for the Entertainment content area:

Item Description:
Movie review of new release, 13th Warrior. Would include image and a text review.

Who should be allowed to add this content?
Member of the movie team

Who should be allowed to view this content?
Everyone

Who should control who can add/view this content?
The head of the movie department

Possible Classifications:
movie, review, adult content, movie rating, kind of movie: action/adventure

Mark the core content for this item:
File_____ Text Item ☒ URL_____ Folder Link_____

PL/SQL Call_____ App. Component_____ Image Map_____ Other:_____

Standard Attributes for Items:
Name, Title, Description, Category, Perspectives (multiple), Author, Expiration

Date, Image, Rollover Image, Keywords

Additional Attributes (may be displayed with item and used for searching):
Review date, movie release date, date available on video

When you find yourself unable to come up with any more items that need different classifications or attributes, you're ready for the next step.

Step 3 Fill out the User Community Worksheet

Use the user community worksheet to organize the information on the item worksheets.

- In the Content Providers column, list your answers to the question "Who should be allowed to add this content?" from the item worksheets.
- In the Owners column, list your answers to the question "Who should control who can add/view this content?" These users or groups of users are most likely to be your folder owners. Folder owners can manage content as well as specify folder security.
- In the Viewers column, list your answers to the question "Who should be allowed to view this content?"

It is quite likely that you will have multiple values in each of the cells of the user community worksheet. For example, you could have two or three different groups of content providers and five or six different groups of viewers for any one folder owner. In the sample item worksheet, the content provider was the movie team, the owner was the movie department head, and the viewers were everyone. It is possible that the movie team could be further divided in a manner to match the sub-folders in the Movies folder, or that these sub-folders would simply represent a more detailed level of classification that is best implemented via folders.

Step 4 Plan your folder structure

You are now ready to start planning your folder structure. The key choice you must make is what to implement as a folder, a category, and a perspective. The best way to tell the difference between a folder and a category is that folders are used by only part of your user community, while the categories make sense for most (if not all) of the user community. Folders and categories should both be specific in nature. It should be easy to tell to which folder branch and category each item belongs. Perspectives are those classifications that are harder to separate. Review the perspectives for the sample Entertainment content area in Table 2–2, "Entertainment Content Area Categories and Perspectives" again. A movie review for a movie

might be in the Family Friendly, Full Color, and 1980s perspectives, but will only make sense in one category, the Review category.

Keeping all of the above information in mind, review your user community worksheet again. When a single entry in the Owners column corresponds to multiple entries in the Content Providers column, it typically signifies a branch in your folder hierarchy. Review your item worksheets again, paying particular attention to the possible classifications. Note any classifications that you feel correspond to the owners and content providers you have listed on the user community worksheet. If a classification corresponds to an owner or content provider, it is probably a good candidate to become a folder or subfolder.

In the Entertainment content area example, our sample item listed both Reviews and Movies as possible classifications. Looking at our user community, we see that while all the information managed by the head of the movie department has to do with movies, only some of it is classified as reviews. By looking at the rest of our Entertainment content area structure, you can see how the Review category would be listed as a possible classification for many sample items and in many different parts of the content area, while a movie classification would only be used in the movie folder and its subfolders. Therefore, it makes sense to make Review a category and Movies a folder.

2.2.2 Information Analysis

Once you have gathered information about the kinds of items that will make up your content area and the users who will build and view the content, it is time to carefully consider the specific attributes of your content area.

Step 1 Consider Security

The first step in creating a content area is to think about which members of your user community need privileges to modify the content and appearance of the content area. You have already started this process by gathering content area organizational information in the user community worksheet.

The basic building block of a content area is the folder. Folders in Oracle Portal are very similar to folders in a file system, but they also form the basis for content area pages. For example, when you click on a link to a folder, a page listing the contents of the folder is displayed. Like file system folders, Oracle Portal folders are hierarchical. Each folder has its own security and visual style properties, which may or may not be inherited from their parent folder. As discussed in Section 2.2.1, "Gathering Information", the user community worksheet can give you a good start in designing the folder structure of your content area.

Since most security is applied at the folder level, your folder structure will usually mirror the organization of the people who will maintain and access your content area. While you can implement item-level security with Oracle Portal, it typically requires less work to administer security at the folder level than at the item level—that is, to give users privileges on folders rather than individual items.

In our Entertainment content area, the folder structure has been designed for a group of content providers who are specialists in a particular area of entertainment. For instance, the owner of the Dance folder has complete control over the subfolders of the Dance folder. The owner of the Music folder, therefore, should not be able to add information to the Dance folder because Dance is not his or her specialty.

Imagine for a moment that the folder structure was organized differently. Imagine that the folders and categories were reversed, and all of the categories were folders and vice versa. The Dance specialist and the Music specialist would both need to have some control over the Review folder. As a result, access privileges would probably have to be maintained for each item. If the Review folder contained hundreds of reviews, this would quickly become a cumbersome administrative task. It is far better, then, to have folders that correspond to the different groups of content providers. In the case of our Entertainment example, this reversed folder-category structure would make sense only if the content providers were divided by the *kind of information* they knew about (reviews, venues), rather than the *type of entertainment* (movies, dance).

The most important thing to remember about folder organization is that folders are the most efficient means for assigning access privileges. Regardless of the folder structure, the viewer of the Entertainment content area can easily access a page of music, dance, and theater reviews.

Step 2 Consider Your Content

Oracle Portal content areas are especially good for classifying the items and other content that you add to them. The two main mechanisms for classifying your items are categories and perspectives. The key difference between these two mechanisms is that an item must belong to only one category, but can belong to zero or more perspectives.

Categories are best thought of as answering the question, "What is this item?" Since each item must belong to one and only one category, categories should be very basic classifications. Refer back to your item worksheets and consider the kinds of information that your content providers will be adding to your content area. Since content providers must assign every item to one category, the best categories are

very distinctive and make it easy for your content providers to know which one to choose. If you find that your categories overlap, you should rethink the structure.

You can also create categories in a hierarchy. That is, you can create a parent category and group similar categories under it as subcategories. This hierarchical organization can be useful when you have a large number of categories and perspectives--but be careful not to create too many of them to avoid confusion. If you have more than ten categories, users may be overwhelmed by the options and choose the first category that roughly fits rather than the most apt category. In the Entertainment content area, you can see how the Review category is a very distinct classification that is useful across many of the folders. The harder it is for your content provider to choose a category for their content, the more likely it is that they will incorrectly classify their content, thus making it difficult to find.

Perspectives are best thought of as answering the questions, "Who will be interested in this item?" or "What is the role of the person to whom this item is the most important?" Perspectives can highlight special attributes of an item. In the Entertainment content area, we used the Outdoors perspective to identify items that describe something outdoors. A user scanning the contents of one of the Music folders will be able to see right away which items are about outdoor performances or outdoor venues.

Review your item worksheets again. Then, fill out the category and perspective worksheets below to sort through the remaining classifications for items that are not folders. It may be helpful to circle the folder for each item as it is listed in the possible classification portion of your worksheet. Your best categories are often those classifications that first seemed like they would be folders but were not used for the folder structure. Remember: both categories and perspectives are most powerful when they are usable across many different folders in the content area.

Figure 2–3 *Candidate Category Structure: Content Area Name:* _____

Category Name	Category Description	Child Categories	Example Items

Figure 2–4 *Candidate Perspective Structure: Content Area Name:* _____

Perspective Name	Description	Child Perspectives	Example Items

Step 3 Consider the User's Point of View

As you design your content area, it is important to keep in mind how it will be used. Users who are looking for the information in your content area have no idea that the content is organized in folders, categories, and perspectives. They simply click on the name of a category (that is, a category link), and it instantly takes them to a list of items assigned to the selected category (excluding any items to which they do not have viewing privileges). Likewise, clicking on the name or image of a perspective (that is, a perspective link) takes the user to a list of items in the content area assigned to that perspective.

Think about where you want to place your category and perspective links. A category link may be displayed:

- on a navigation bar
- as a group banner on a page above a list of items
- as criteria for an advanced search operation

Perspective links can appear in the same places as category links. In addition, you can optionally expose a perspective link next to each item assigned to the perspective by selecting the Display Icon With Items property in the Perspective Manager.

Step 4 Consider Navigation Aids

Folders, categories, and perspectives enable you to organize and cross reference your items in such a way that it is easy for your end users to find information in the content area.

Examine the top two levels of your folder structure. These are the folders that will be initially displayed on the default main content area page. Verify that the folder names are distinct from one another. Your top level branches in the folder structure should not have names that are easily confused with one another.

Filling out the Folder Tree worksheet below will give you an idea of what your top two levels of folders will look like to a user.¹ Watch out for folders that are too similar. If the folders are not distinct, users will struggle to distinguish between them, making it difficult to know where to put and look for information.

¹ Keep in mind that not every user will have privileges to see every folder. Given that, you may want to make copies of the folder tree worksheet for different groups of users.

Figure 2-5 Folder Tree Worksheet

[illegible]

After you have built your content area, examine the content area page as a whole (e.g., the navigation bar and folder listing). Sometimes users are looking for something very specific and at other times they are browsing to determine what information is available. Some users like to jump directly to an item, while others prefer to navigate to items sequentially. Your content area page should serve all of these scenarios. It is common to include a search box on the navigation bar for those users who would rather search than browse folders. Users who want to browse can make use of the folder structure, categories, and perspectives.

You should also consider issues such as whether you want the navigation bar to be a persistent part of the content area's interface. Oracle Portal enables you to use the same navigation bar throughout the content area, use different navigation bars depending on the context, or remove the navigation bar altogether. If you choose to have navigation bars, consider where to place them on the page (e.g., left, right, top, or bottom) and what to include in them. Navigation bars can contain a wide variety of elements (e.g., basic search, hyperlinks, and pull down lists). If you choose not to have a navigation bar, then you should include other mechanisms for simplifying navigation within the content area.

Step 5 Consider item types

When users add to your content area, they must identify the type of item (e.g., file or URL). Oracle Portal content areas come with an assortment of default item types. You can add to these default types as necessary. The default item types are:

- File
- Folder Link
- Image
- Java Application
- PL/SQL
- Text
- URL
- Application Component
- Zip File

Once users identify the item type, they are given a list of attributes to define for that item. This table helps you understand the attributes for each of the default item types:

Table 2–3 Attributes by Default Item Type (Required)

Attribute	File	Folder Link	image	Java App	PL/SQ L	Text	URL	App. Comp.	Zip File
Application Component								x	
Application Type				x					
Category	x	x	x	x	x	x	x	x	x
Description	x	x	x	x	x	x	x	x	x
Display Name	x	x	x	x	x	x	x	x	x
Expiration Period	x	x	x	x	x	x	x	x	x
File Name	x								x
Image			x						
Imagemap			x						
Imagemap Name			x						
Initial Page Name				x					
Jar File				x					
Path		x							
PL/SQL Code					x				
Publish Date	x	x	x	x	x	x	x	x	x
Text						x			
URL							x		

Table 2–4 Attributes by Default Item Type (Optional)

Attribute	File	Folder Link	Image	Java App	PL/SQL	Text	URL	App. Comp.	Zip File
Author	x	x	x	x	x	x	x	x	x
Basic Search Keywords	x	x	x	x	x	x	x	x	x
Display Options	x				x	x	x	x	
Enable Item Check-Out	x	x	x	x	x	x	x	x	x
Hide Item	x	x	x	x	x	x	x	x	x
Image	x	x	Req.	x	x	x	x	x	x
Image Alignment	x	x		x	x	x	x	x	x
Perspectives	x	x	x	x	x	x	x	x	x
Schema					x				

You cannot alter or add to the attributes defined for the base item types. If you wish to append additional attributes to a default item type, you can create a custom item type. Custom item types based on a default item type inherit the default's attributes, plus any custom attributes you create. In this way, you can use custom item types to collect additional structured information about items via their attributes.

You can also create custom item types from scratch, without basing them on a default type. In this case, the item type begins with only the most basic attributes, such as Display Name, Category, and Perspectives. You must add any other desired attributes to the item type yourself.

Review your item worksheets again, focusing on what you entered in the Additional Attributes column. If you don't have any additional attributes, then the default attributes will provide all the information necessary to track your items. If you listed additional attributes, use the item type worksheet below to figure out how many additional item types you need for your content area. Keep in mind that you can reuse attributes that you create across item types. For example, if you add a Due Date attribute and use it in two different item types, a search for items with a specific due date will examine the Due Date attribute for items in both item types.

List all of the additional attributes in the column below Attribute Name. List a data type for each additional attribute (such as number, date, character, file, URL, etc.). In the cell beneath each Item Type Name label, enter a name for your new collection of

attributes. In the cell beneath each Based On label, enter the name of a default item type to indicate the item type to which you want to add your attributes. Mark with Xs those attributes used in each item type.

When you want to associate additional attributes with the folder instead of the item, you can create a custom folder type.

2.2.3 Troubleshooting Your Content Area Structure

Even the most diligent planners cannot always foresee every contingency. Hence, you will undoubtedly run into issues after you have created your content area that will cause you to revisit its organization. These sections describe some common problems that arise after you have implemented your content area.

2.2.3.1 All of My Content Areas Have the Same Categories

Oracle Portal provides a Shared Objects content area where you can store categories, perspectives, styles, navigation bars, attributes, and item and folder types that apply to all of your content areas. If you find that you are constantly recreating the same things for every content area, you should consider building them in the Shared Objects content area so that you can build them once and reuse them in multiple content areas. Shared Objects can increase the integration and consistency between content areas.

2.2.3.2 Item Level Security is Required Everywhere

You may be required to use item level security for your content area, but this is often a sign that you could have set up your content area differently. The main thing to keep in mind is that folders should be the main level at which security is applied. Look at your categories. Would your site work better if your categories were folders and your folders were categories?

2.2.3.3 I Do Not See Links to My Perspectives

You may need to change your settings to ensure that you have links to your perspectives.

- To display perspective links next to items, set the **Display Icon With Items** property in the Perspective Manager and update your style to include perspective links with regular items (use the Folder Layout tab of the Style Manager).
- To add perspective links to your navigation bar, use the Navigation Bar Manager to add either individual links/images or a drop down list that enables the user to select a perspective to view.
- To simplify searching for perspectives, add a folder based upon a saved search. Give the saved search folder the same name as the perspective and select the

perspective you would like to highlight in the search screen. This will limit the search to look within the content area at all items belonging to the chosen perspective.

2.2.3.4 My Categories Do Not Appear in the Content Area Map

Consider creating a folder based upon a saved search. You can add a folder to your hierarchy and define it as a search for all items belonging to a specific category. If you name that folder the same as your category, it will seem to users that there is a folder on the content area map holding all of the items in that particular category.

One advantage of this type of folder is that it is completely dynamic. As new content is assigned to the category, the search picks up the new items without any intervention from you. This method works just as well for any type of search that you want to save and make available to your users.

Creating an intranet portal and adding simple portlets

This chapter describes how to use Oracle Portal to build an interface, or a portal, that contains such material as calendars, reports from your own database, links to external Web sites, and customized HTML. If you've already built several portals and wish to work with advanced applications, see Chapter 4, "Creating a line of business page for your intranet portal".

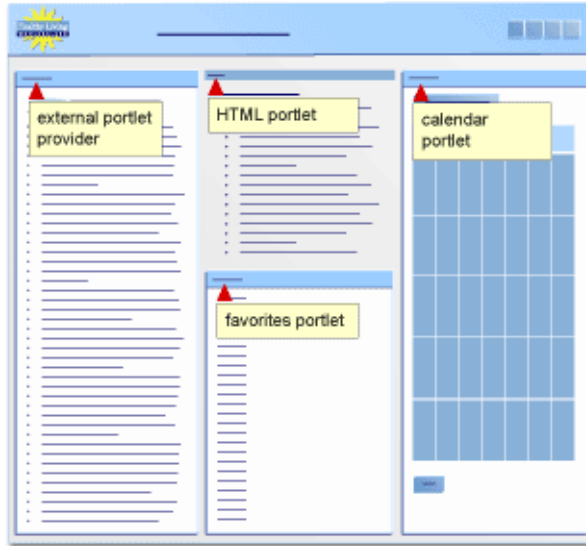
In this chapter, you will create a portal that enables users to access multiple components and information from the same screen. The first step in creating a portal is designing the **page**: figuring out what applications your users need, what information they would like to see, and how you want the information to display. For more information on planning a page, see Chapter 2, "Planning Your Web Content".

You will then set up the page for your portal (the screen that displays the applications, links, etc.) with a style you like, create the applications you want to include in the portal, then add those applications to the portal so users can easily access them.

To build a simple intranet portal, you will use the Create Page Wizard to create a page and the Customize Page tab set to add and edit portlets. You will also use the Create Calendar, Create Forms, Create Sequence, and Create Links wizards to build an event calendar.

You need Edit or higher access privileges to the demo schema to complete this chapter. You also need Edit or higher application privileges to build the calendar portion of this chapter. If you do not have these privileges, contact your portal administrator.

The following figure illustrates the four parts of your simple intranet portal.



The completed portal is available in the installation package that contains the demo schema. For more information, contact your portal administrator.

This chapter contains the following sections:

Feature	Location
Use the Create Page Wizard to create a page.	Section 3.1, "Create a page"
Create an HTML portlet that contains your own customized HTML content.	Section 3.2, "Add customized HTML code as a portlet"
Add and modify a Favorites portlet.	Section 3.3, "Add and customize a Favorites portlet"
Use the Create Calendar Wizard, the Create Link Wizard, the Create Form Wizard, and the Create Database Sequence Wizard to build an updateable calendar portlet.	Section 3.4, "Create an event calendar and add it as a portlet"
Display the portal to the public.	Section 3.5, "Displaying the portal to others"

3.1 Create a page

The steps in this section show you how to use the Create Page Wizard to create a page, and design its layout.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Page Tasks** and click **Go**. In the right pane, click **Creating a page**.

1. Log into Oracle Portal.

Note: If you do not have a user name and password, contact your portal administrator.

2. In the Navigator, click the **Pages** tab.
3. Next to My Pages, under Actions, click **Create**.

The Page Properties tab of the Create Page Wizard displays.

4. In the **Name** field, enter <YourName>_Main.
5. In the **Display Name** field, enter <YourName> Main.
6. Click **Next**.
7. From the **Layout** drop-down list, choose **Plain Layout**.

A preview of the layout displays below the list.

8. From the **Style** drop-down list, choose **Healthy Living Page Style**.

A preview of the style displays below the list.

9. Click **Next**.
10. Click **Next**.
11. Select the **Publish As Portlet** check box.
12. Click **Finish**.

Tip: If you end your Oracle Portal session, you can always return to this page by going to the Navigator. Click the **Pages** tab, then **My Pages**. Then, find the Main page. Click any of the actions listed next to the Main page, such as View or Customize.

13. The Main page now displays in your browser.



Note: You can change the graphic in the banner by clicking the **Edit Defaults** link in the banner located on the Portlets tab of the Customize Page dialog. Under Banner Logo Settings, enter the filename of the logo you want to use.

The page should look something like this:



3.2 Add customized HTML code as a portlet

The steps in this section help you to add customized HTML as a portlet. When you create an HTML portlet, you can write your own HTML in a text editor and paste it in a portlet, or write HTML directly into the portlet. You can use this portlet when you have HTML you want to copy into your portal, for example a banner. In this section, we create a company news HTML page and paste it into the portlet.

1. In the Navigator, click the **Pages** tab.
2. Navigate to the **<YourName> Main** page.
3. Next to the page, click **Edit**.
4. On the Portlets tab, in the middle column, click .
5. In the **View By** list, make sure **Provider** is chosen.
6. Under Oracle Portal, find **HTML Portlet**.
7. Click  to add the portlet.
8. Click **OK**.
9. In the HTML portlet, click **Edit Defaults**.
10. In the **Display Name** field, enter Healthy Living Company News.
11. In the **Content** field, paste the following HTML:

```
<html>
  <li>Nutritionists choose Healthy Living as their #1 wholesaler.</li>
  <li>Healthy Living allies with so-far-unnamed wholesale healthy pet food
company.</li>
  <li>New associate program now available. Become an affiliate of Healthy
```


Living!</body></html>

12. Click **OK**.

13. Click **Close**.



Your portal should now look something like this:



3.3 Add and customize a Favorites portlet

The steps in this section help you customize the Favorites portlet by adding a new favorite link and deleting existing links. You can use the Favorites portlet to maintain and organize your more commonly used services. For example, instead of relying on the bookmarks in your browser, you can add the same links to the Favorites portlet and access the Web sites via your portal.

3.3.1 Add the Favorites portlet

1. Display the Portlets tab of the Customize Page dialog for the Main page.
2. On the Portlets tab, in the middle region, click .
3. In the **View By** list, make sure **Provider** is chosen.
4. Under Oracle Portal, find **Favorites**.
5. Click  to add the portlet.
6. Click **OK**.

7. Click **Close**.
8. Click **Finish**.

3.3.2 Add a new favorite link

When you add the Favorites portlet, the Oracle Portal default links display. To take advantage of this portlet, you will always want to add new links to this portlet. You can enable your users to add and remove links, but you might want to add a few commonly-used ones. For example, you might add your organization's Internet address.

1. On the Main page, in the Favorites portlet, click **Customize**.

Tip: To display the Main page, click the **Pages** tab in the Navigator. Click **My Pages**, then find <YourName>_Main page.

2. In the Customize Favorites dialog, click **Add or Edit Favorites**.
3. On the Navigate Favorites page, click **Create Favorite**.

The Add Favorite page displays.

4. In the **Name** field, enter `Oracle Technology Network`.
5. In the **URL** field, enter `http://technet.oracle.com`.
6. Click **Create**.

Oracle Portal creates the new link and displays the Add Favorite page again so you can add another favorite link.

Note: You can click the link at the top of the page to edit the link you just created.


7. Click **Close**.

The new Oracle Technology Network favorite now displays on the Navigate Favorites page.

The Oracle Technology Network link should now display in the Favorites portlet.



3.3.3 Remove an existing favorite link

1. On the Main portal, in the Favorites portlet, click **Customize**.
2. Click **Add or Edit Favorites**.
3. Next to Oracle (for example), click **Delete**.
4. In the upper right corner, click **Close**.
5. In the shortcut bar, click .
6. On the Pages tab, click **My Pages**.
7. Click **Main**.

The Favorites portlet should now look like this:



3.4 Create an event calendar and add it as a portlet

The steps in this section help you create an event calendar, where you can add an event and display it in a calendar format. One of the most common portal functions is a calendaring application, where users can immediately see the events for a particular day.

In these steps, you will create an Oracle Portal application. To do so, you will use the following:


- n Create Calendar Wizard to create the calendar
- n Create Forms from Table or View Wizard to create a form that updates the calendar
- n Create Sequence Wizard to create a database sequence for the information the form accepts
- n Create Links Wizard to connect the form and the calendar
- n Create Calendar Wizard again to create a button to access the form.

Note: To create the components in this section, you must have access to the <DEMO SCHEMA>.EVENTS table, where DEMO SCHEMA is the name of the schema where the demo schema objects are installed. You must also have access to create components within the Casebook30_Application application. If you do not have access to this table, or if you do not have access to this application, contact your portal administrator. Instructions are in Appendix A of this manual.

3.4.1 Create an event calendar using the Create Calendar Wizard

To create a view-only calendar, where users can view the dates but not add events, you can simply follow the steps in this section (Section 3.4.1). To make the calendar updateable, however, you will need to complete all sections in Section 3.4, "Create an event calendar and add it as a portlet".

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Application Tasks** and click **Go**. In the right pane, click **Building Components**.

1. In the shortcut bar, click .
2. Click the **Applications** tab.
3. Make sure the path looks like this: Path: Applications. If you do not see this exact path, click **Applications** to access the root folder.
4. Under Name, click **Casebook30_Application**.

You should now see the following path: Path: Applications, Casebook30_Application.

5. Next to Create New, click **Calendar**.

The Create Calendar Wizard displays.

6. In the **Calendar Name** field, enter <YourName>_EVENT_CAL.
7. In the **Display Name** field, enter <YourName> Event Calendar.

8. In the Application list, choose **Casebook30_Application**.
9. Click **Next**.
10. On step 2 of the Create Calendar Wizard, enter the following SQL query in the **SQL Query** field:

```
select
    e.event_start_date    the_date,
    e.event_title         the_name,
    null                 the_name_link,
    null                 the_date_link,
    null                 the_target
from <DEMO SCHEMA>.hlw_events e
order by 1
```

Note: Do not directly copy and paste the SQL from this text. You must replace the schema name (DEMO SCHEMA) with the name of the schema where the demo schema objects are installed.

11. Click **Finish**.

Note: Doing so accepts the defaults in the remainder of the wizard. You can always return to the wizard later to customize the appearance of the calendar.

12. The Manage Component dialog for the new calendar displays.

Note: You can always view the Manage Component dialog for a particular component by clicking **Manage** next to the component name on the Application tab.

13. Click the **Access** tab.
14. Select the **Publish to Portal** check box.
15. Click **Apply**.
16. Click the **Develop** tab, then click **Run**.

The calendar displays in a new browser window, and should look something like this:



Note: If you get the error "No Rows Returned," it's because you have no data in the calendar. Continue the steps in this section. After you've created the form, you can insert data, and the calendar should display.

3.4.2 Create a form to update the calendar

To manage events using the calendar, you'll need to create a form in which users can add, remove, and modify events.

Note: To complete this section, you must have UPDATE privileges on the <DEMO SCHEMA>.HLW_EVENTS table. If you do not have these privileges, contact your database administrator.

1. In the Navigator, click the **Applications** tab and navigate to **Casebook30_Application**.

Tip: Follow steps 2 through 5 in Section 3.4.1, "Create an event calendar using the Create Calendar Wizard".

2. Next to Create New, click **Form**.
3. On the Forms page, click **Forms based on table or view**.
The Create Form from Table or View Wizard displays.
4. In the **Name** field, enter <YourName>_EVENT_FORM.
5. In the **Display Name** field, enter <YourName> Event Form.

6. From the Application list, choose **Casebook30_Application**.

7. Click **Next**.

8. In the **Table or View** field, enter `<DEMO_SCHEMA>.HLW_EVENTS`.

Note: DEMO_SCHEMA is the name of the schema in which the demo schema objects are installed. The table now displays in the **Table or View** field.

9. Click **Next**.

10. On the Form Layout tab, make sure **Tabular** is selected.

11. Click **Next**.

12. In the left pane, make sure **FORM** is highlighted.

13. In the right pane, scroll down until you see the PL/SQL text box.

14. In the text box, enter something like this:

```
go(' <http://<host name>/pls/<DAD>/<DEMO_SCHEMA>.<YourName>_EVENT_
CAL.show> ');
```

Note: Replace the text "`<host name>`" with the URL for Oracle Portal, ex. "`http://mycomputer.domain.com/pls/portal30`". You can find this URL by displaying the Manage Component dialog for the `<yourName>_EVENT_CAL` and dragging your mouse over **RUN**. The URL displays at the bottom of your browser.

15. Click **Finish**.

Doing so builds the form using the default values in the remaining steps of the wizard. If you want to see the default values, click **Next** in each of the wizard steps, but do not change any of the values. Then, click **Finish** when you're done.

16. When you click **Finish**, the Manage Component dialog for your new form displays.

17. Click the **Access** tab.

18. Select the **Publish to Portal** check box.

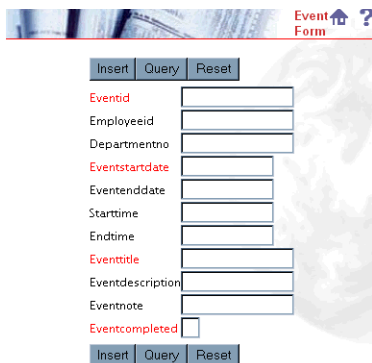
Note: Selecting this check box enables you to display the form as a portlet.

19. Click **Apply**.

20. Click the **Develop** tab.


21. Click **Run**.

The new form displays in a separate browser window.

The screenshot shows a web application interface for an 'Event Form'. At the top, there is a header bar with a blue background and a red border. On the right side of the header, the text 'Event Form' is displayed in red, followed by a small blue house icon and a question mark icon. Below the header, there are three buttons: 'Insert', 'Query', and 'Reset', each with a blue background and white text. The main area of the form contains several input fields, each with a red label to its left. The labels are: 'Eventid', 'Employeeid', 'Departmentno', 'Eventstartdate', 'Eventenddate', 'Starttime', 'Endtime', 'Eventtitle', 'Eventdescription', 'Eventnote', and 'Eventcompleted'. The 'Eventcompleted' field is a checkbox. At the bottom of the form, there are three buttons: 'Insert', 'Query', and 'Reset', each with a blue background and white text.

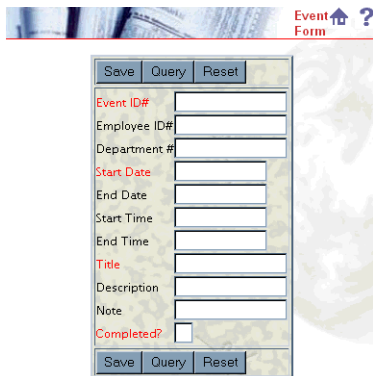
3.4.3 Change the appearance of the form

Since users of your calendar will also use the form to update and view events in the calendar, you might also want to customize the way the form displays.

1. On the **Applications** tab, next to <YourName>_EVENT_FORM, click **Edit**.
2. On the Formatting and Validation Options page, make sure FORM is highlighted.
3. Under Form Level Options, choose **Wheat** from the Box Background Color drop-down list.
4. Choose **Dude** from the Box Background Image drop-down list.
5. Choose **Thick Border** from the Box Border drop-down list.
6. Order the query by **EVENT_START_DATE**.
7. From the list, choose **Ascending**.
8. In the left pane, if the FORM item list is collapsed, click .
9. Click **INSERT_TOP**.
10. In the right-hand pane, in the Label field, enter Save.
11. In the left-hand pane, click **INSERT_BOTTOM**.
12. In the right-hand pane, in the Label field, enter Save. Completing steps 9-13 changes the button label to Save in the form.
13. In the left-hand pane, click **EVENT_ID**.
14. In the right-hand pane, in the Label field, enter Event ID#.

15. In the left-hand pane, click **EMPLOYEE_ID**.
16. In the right-hand pane, in the Label field, enter Employee ID#.
17. In the left-hand pane, click **DEPARTMENT_NO**.
18. In the right-hand pane, in the Label field, enter Department#.
19. In the left-hand pane, click **EVENT_START_DATE**.
20. In the right-hand pane, in the Label field, enter Start Date (DD-MON-YY).
21. In the left-hand pane, click **EVENT_END_DATE**.
22. In the right-hand pane, in the Label field, enter End Date (DD-MON-YY).
23. In the left-hand pane, click **EVENT_START_TIME**.
24. In the right-hand pane, in the Label field, enter Start Time.
25. In the left-hand pane, click **EVENT_END_TIME**.
26. In the right-hand pane, in the Label field, enter End Time.
27. In the left-hand pane, click **EVENT_TITLE**.
28. In the right-hand pane, in the Label field, enter Title.
29. In the left-hand pane, click **EVENT_DESCRIPTION**.
30. In the right-hand pane, in the Label field, enter Description.
31. In the left-hand pane, click **EVENT_NOTE**.
32. In the right-hand pane, in the Label field, enter Note.
33. In the left-hand pane, click **EVENT_COMPLETED**.
34. In the right-hand pane, in the Label field, enter Completed?.
35. Click **OK**.
36. In the Manage Component dialog, click **Run**.

The form now displays in a new browser window with the new colors and label names.




The screenshot shows a web form titled "Event Form" with a home icon and a question mark. The form contains the following fields: Event ID#, Employee ID#, Department #, Start Date, End Date, Start Time, End Time, Title, Description, Note, and a Completed? checkbox. There are "Save", "Query", and "Reset" buttons at the top and bottom of the form.


3.4.4 Create a database sequence for the form

Since you will be allowing users to add, update, and delete events by using your new form, you will need to assign sequence numbers to each record. You can use the Create Sequence Wizard to enable Oracle Portal to automatically generate a new sequence number for each record in the database. This way, when a user edits or removes an event, the user (and Oracle Portal) can refer to the sequence number to retrieve the appropriate record.

Note: You must have `MANAGE ALL SCHEMAS` global privilege or `Manage` privileges on the `<DEMO_SCHEMA>` schema to create database objects.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Administration Tasks** and click **Go**. In the right pane, click **Building Database Objects**.

1. In the shortcut bar, click .
2. In the Navigator, click the **Database Objects** tab.
3. Next to Path, make sure it says Path: Database Schemas, `<DEMO_SCHEMA>`. If you do not see this, click **Database Schemas** to navigate to the root of the directory. Then, find `<DEMO_SCHEMA>` in the list. Click `<DEMO_SCHEMA>` to access the schema.
4. Next to Create New, click **Sequence**.
5. On step 1 of the Create Sequence Wizard, leave the **Schema** name as `<DEMO_SCHEMA>`.
6. In **Sequence Name**, enter `<YourName>_EVENT_SEQ`.
7. Click **Next**.

8. On step 2, in the **Increment By** field, enter 10.
9. In the **Start With** field, enter 1000.
10. Click **Next**.
11. On step 3, click **Finish**
`<YourName>_EVENT_SEQ` now displays in the Database Navigator.
Note: If you don't see the sequence in the list, you may have to click **Next**, or you can use the **Find** field.
12. In the Database Objects Navigator, next to `<YourName>_EVENT_SEQ`, click **Grant Access**.
13. In the **User/Role** field, enter PUBLIC.
Tip: Granting access to PUBLIC enables the form you create to use this sequence. Doing so also enables anyone who can use your form to insert records in the database. Note that you can also grant access on a user-by-user or group-by-group basis.
14. From the list, choose **SELECT**.
15. Make sure the **with Grant Option** check box is not selected.
Note: If you select the **with Grant Option** check box, you will enable the public to grant other users privileges to the database sequence.
16. Click **Add to List**.
17. Click **Close**.
18. Click the **Applications** tab.
19. In Casebook_HLW_APP, find `<YourName>_EVENT_FORM`. Note that the path should display as follows: Path: Applications, Casebook30_Application, Components.
20. Next to `<YourName>_EVENT_FORM`, click **Edit**.
21. On the Formatting and Validation Options page, in the left-hand pane, find the `EVENT_ID` column. You may have to click  next to FORM to display all the columns.
22. Click `EVENT_ID`.
23. In the right-hand pane, find the **Validation** section.
24. In the Default Value field, enter:
`#<DEMO_SCHEMA>.<YourName>_EVENT_SEQ.nextval`
25. From the Default Value Type list, choose **Expression returns number**.

26. Click OK.

Now, whenever a user adds an event using the EVENT_FORM, Oracle Portal automatically assigns the new record an ID number. This number is unique to each record, and makes it easy to retrieve, update, and delete the record.

3.4.5 Create a link between the form and the calendar

To enable users to update the event calendar, you must create a link between the form and the calendar.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Application Tasks** and click **Go**. In the right pane, click **Building Components**, then click **Using a link between components**.

1. On the Applications tab, next to Create, click **Link**.
2. On step 1 of the Create Link Wizard, in the **Link Name** field, enter EVENT_LINK.
3. In the **Application** list, choose **Casebook30_Application**.
4. Click **Next**.
5. On step 2 of the Create Links Wizard, select **Oracle Portal Component**.
6. In the **Target Component or URL** field, enter Casebook30_Application.<YourName>_EVENT_FORM.
7. Click **Finish**.

The link now displays in the Manage Component dialog.

8. Click **Run**.
9. In the Navigator, click the **Application** tab and navigate to <Your Name> EVENT CAL.
10. Click **Edit**.

The SQL Query for your event calendar displays.

11. In your SQL query, add your link to the the_name_link column of the query. To do so, edit the SQL query so that it looks something like this:

```
select
    e.event_start_date the_date,
    e.event_title      the_name,
    <DEMO_SCHEMA>.www_app_module.link?p_arg_names=moduleid&p_arg_
values=<EventFormID>&p_arg_names=_event_id_cond&p_arg_values=%3D&p_arg_
names=_show_header&p_arg_values=YES' || E.EVENT_ID || '
```

```

        the_date_link,
        null          the_target
from <DEMO_SCHEMA>.events e
order by 1

```


Note: Do not copy and paste the code directly from this text. You should change the text from DEMO_SCHEMA to the user name that owns the schema, as well as change the name of the application to your application. Also, replace "EventFormID" with the module ID of <YourName>_EVENT_FORM. You can find this value by displaying the Manage Component dialog for <YourName>_EVENT_FORM. Drag your mouse over **Run**, and the module ID displays at the end of the URL at the bottom of your browser. See your portal administrator for more details.

12. Click **OK**. The calendar displays in the Manage Component dialog.

You have now finished creating links for your calendar.

3.4.6 Create a button on the calendar to access the form

To make the form conveniently accessible, you need to create a button on the calendar.

1. On the **Application** tab of the Navigator, next to <YourName>_EVENT_CAL, click **Edit**.
2. Click  to display the Additional PL/SQL Code tab.
3. In the Calendar column, find the **...before displaying the header** field.
4. Enter Javascript to create a button called "New Appointment" that, when clicked, executes the form you created in Section 3.4.3, "Change the appearance of the form".

To do so, in the field, enter:

```

http.formOpen(owa_util.get_owa_service_path|| '<PORTAL_SCHEMA>.wwa_app_
module.link?p_arg_names=_moduleid&p_arg_values=<EventFormID>&p_arg_names=_
sessionid&p_arg_values=&p_arg_names=EVENT_START_DATE&p_arg_
values=%23sysdate&p_arg_names=_event_start_date_cond&p_arg_values=%3D&p_arg_
names=_show_header&p_arg_values=YES');
http.formSubmit(null, 'New Appointment');
http.formClose;

```

Note: Replace "EventFormID" with the module ID of <YourName>_EVENT_FORM. You can find the module ID for a component by display its Manage Component dialog, then dragging your mouse over **Run**. The module ID displays at the bottom of your




browser. Copy the code in the calendar column in the ‘... after displaying the header’ box.

5. Click **OK**.
6. In the Manage Component dialog, click **Run**.
7. Click **New Appointment**.
8. In the form, enter sample information, making sure you enter information for the required fields (in red).
9. Click **Save**.
10. Click your browser’s Back button to return to the calendar.
11. Click your browser’s Reload or Refresh button.



The event now displays in the calendar.

3.4.7 Add the calendar as a portlet

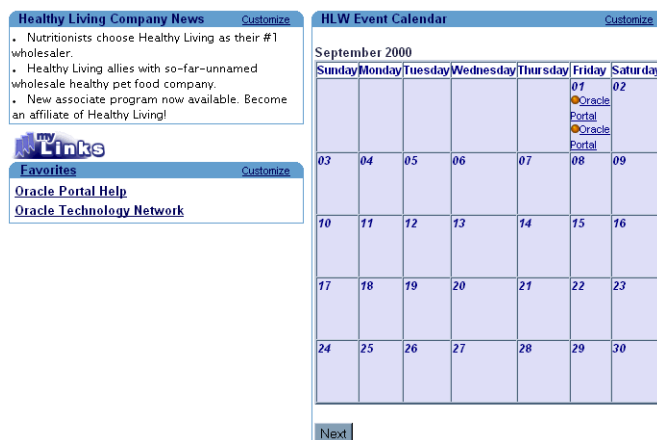
Now that you’ve created a calendar that users can update, you can add the calendar directly to your intranet portal.

1. In the shortcut bar, click .
2. In the My Pages list, next to <YourName>_Main, click **Edit**.
3. In the right region, click  to add a column.
4. In the new region, click  to add a portlet.
5. In the **Providers** list, under Casebook_HLW, find <YourName>_EVENT_CAL.

Tip: If your calendar does not display in the drop-down list, return to the Navigator and find the event calendar. Next to the calendar, click **Manage**. On the Access tab, make sure the **Publish to Portal** check box is selected, then click **Apply**.

6. Click  to add the calendar as a portlet.
7. In the **Selected Portlets** list, click <YourName>_EVENT_CAL.
8. Click  to move the calendar above the Favorites portlet.
9. Click **OK**.
10. Click **Close**.


The calendar now displays as a portlet in the right-hand region of your page.



3.5 Displaying the portal to others

Currently, the only users who can view your portal are the DBA and yourself. To make your portal useful to others, you should grant other users access to the portal.

1. In the Navigator, click the **Applications** tab.

Tip: To access the Navigator, click  in the shortcut bar.

2. Navigate to the path: Applications
3. In the list, next to Casebook30_Application, click **Grant Access**.
4. In the **Grantee** field, type Public.
5. From the list, select EXECUTE.
6. Click **Add to List**. Public now displays in the Modify Privilege list.
7. Click **OK**.
8. Test the user's privileges by exiting Oracle Portal, then logging in as the user. When you display the Main portal (in the Navigator, click the **Pages** tab, then find the Main page you created), you should see all the applications you built in Chapter 1.

3.6 Summary

Congratulations! You have created an intranet portal and added different types of portlets to it. You now know how to:

- Create a page
- Create a customized HTML portlet
- Customize the Favorites portlet
 - Create a new Favorite link
 - Remove an existing Favorite link
- Create an event calendar
- Create a form based on a table to update the calendar
- Create a database sequence to update the event records in your database
- Create a link between the form and calendar
- Add a calendar application as a portlet provider

For more information about creating a portal and using portlets, see the *Oracle Portal Online Help*.

Creating a line of business page for your intranet portal

This chapter describes how to use Oracle Portal to build a portal for a Sales department within the Healthy Living Company intranet. You will create components and pages to make it easy for Healthy Living salespeople to access their customer database, quickly peruse product information, and update their own sales information. They will also be able to view their monthly quotas and commission standings relative to the other salespeople in the company.

You will add a page to the Healthy Living intranet portal you created in Chapter 3, "Creating an intranet portal and adding simple portlets". If you did not complete Chapter 3, you can simply create a new portal in this chapter. You will learn how to build somewhat more advanced applications than you learned in Chapter 3. You will learn how to create different views of your data using the Create Report, Create Chart, and Create Form wizards, and display these three views in the same portlet. You will also learn how to tailor your applications to serve a particular line of business.

You need Edit or higher access privileges to the demo schema to complete this chapter. You also need Edit or higher application privileges to build the components in this chapter. If you do not have these privileges, contact your portal administrator.

The following figure illustrates the various parts of the Sales Department portal.



The completed portal is available in the installation package that contains the demo schema. For more information, contact your portal administrator.

This chapter contains the following sections:

Feature	Location
Create a page with tabs using the Create Page Wizard and add an existing page as a portlet.	Section 4.1, "Create a tabbed page and add a page as a portlet"
Create a folder portlet.	Section 4.2, "Add an existing folder as a portlet"
Use the Create Report from SQL Query Wizard to create a sales report portlet, then use the Create Links and Create Chart from Query Wizards to create a chart with links to the sales report.	Section 4.3, "Create a sales chart portlet that links to a report"
Create two charts, then add them to the same portlet.	Section 4.4, "Create a multi-tabbed region containing two charts"

Feature	Location
Create a chart using the Create Chart from SQL Query Wizard, a report using the Report from Query Wizard, and a search form, then add them to a single portlet.	Section 4.5, "Create a multi-tabbed region containing various components"
Create a dynamic LOV that depends on a static LOV and add both to an existing form. Then, arrange the applications you created in the tutorial into a single portlet to provide two views of the same data.	Section 4.5.3, "Create four LOVs for the commission search form"
Use the Create Forms on Tables Wizard to create a customer entry form.	Section 4.6, "Create a customer entry form"
Use the Create Report from Query Wizard to create a product report.	Section 4.7, "Create a product report"
Use the Create Links and Create Menu Wizards to create a menu with links to the customer entry form and product report.	Section 4.8, "Create a menu containing links"

4.1 Create a tabbed page and add a page as a portlet

In this section, you will create the layout for the Sales page of the Inside Healthy Living portal. You will also create tabs on the page and add the Main page from Chapter 3 as a portlet.

Note: In order to add the Main page as a portlet, you must select the **Publish as Portlet** check box on the Access tab of the Customize Main Page dialog.







For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Page Tasks** and click **Go**. In the right pane, click **Creating a page**.

1. Log into Oracle Portal

Note: If you do not have a user ID and password, contact your portal administrator.

2. In the Navigator, click the **Pages** tab.

Note: Next to Path, make sure it reads: **Path: Pages**.

3. Next to **My Pages**, click **Create**.
4. In the **Name** field, enter <YourName>InsideHLW.
5. In the **Display Name** field, enter <Your Name> Inside Healthy Living.
6. Click **Next**.
7. From the Layout Template list, choose **Plain Layout**.
8. From the Style list, choose **Healthy Living Page Style**.
9. Click **Next**.
10. Click  to add a tab to the page.
11. Next to your new tab, called "New Tab," click  to add another tab to the page.
12. On the first new tab, click  to edit the tab name.
13. In the **Display Name** field, enter Main.
14. Click **Apply**.
15. Click **Close**.
16. On the tab called "New Tab," click  to edit the tab name.
17. In the **Display Name** field, enter Sales.
18. Click **Apply**.
19. Click **Close**.
20. Make sure the **Main** tab is active.
21. Click  to add a portlet.
22. Scroll down the **Providers** list until you find **Oracle Portal Pages**.
23. Next to <YourName> **Main Page**, click  to add the page as a portlet.

Note: If you did not complete Chapter 3, add HLW Main Page as a portlet. This is the page we've created for you to add.
24. Click **OK**.

You should now see the layout for the Main page display on the Main tab of your portal.
25. Click **Finish**.

Note: You can change the graphic in the banner by clicking on the **Edit Defaults** link in the banner located on the Portlets tab of the Customize Page dialog. Under Banner Logo Settings, enter the filename of the logo you want to use.

You have created a page containing tabs, and added a page as a portlet. Your portal should look something like this:





4.2 Add an existing folder as a portlet

The steps in this section help you to add an existing **folder** as a portlet. When you add items to a **content area**, you must organize them into folders so that others can access the items via a portal. Once you've created these folders, you can expose them as portlets. Note that you cannot expose an **item** as a portlet.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Content Area Tasks** and click **Go**. In the right pane, click **Working With Folders**.

You can also follow these steps to add a content area as a portlet (just select a content area instead of a folder), but only the information in the root folder and the sub-folders will display in the portlet.

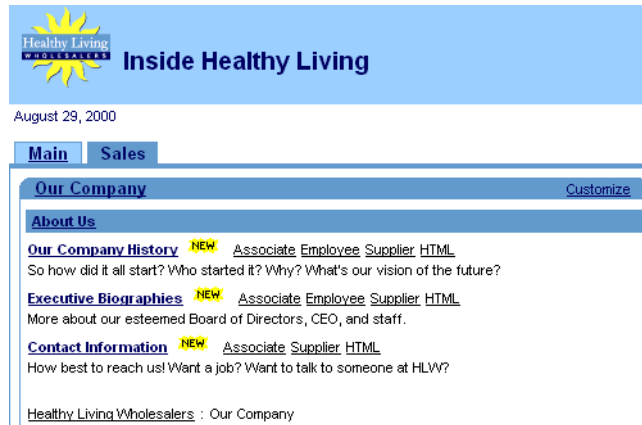
1. In the Navigator, click the **Pages** tab.
2. Navigate to the <YourName> **Inside Healthy Living** page.
3. Click **Customize Page**.
4. Click the **Portlets** tab.
5. Click .
6. In the **View By** list, make sure **Provider** is chosen.
7. Scroll down until you see **HLW**.
8. Next to **Our Company**, click  to add the folder as a portlet.

Note: If you do not see the folder in the list of providers, contact your portal administrator.

9. Click **OK**.

10. Click **Close**.

Your portal, containing the folder, should look something like this:



4.3 Create a sales chart portlet that links to a report

The steps in this section help you create a report using the Query Wizard. Here, you will link a chart to the report, then display the chart as portlet. This way, users can view the chart, then drill down to the report for more information.

If you have not created an application, follow the steps in Section 3.4.1, "Create an event calendar using the Create Calendar Wizard" to create an application under your name. When you create the application, make sure you select the **Expose as Provider** check box on the Access tab for the applications.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Application Tasks** and click **Go**. In the right pane, click **Building Components**.

4.3.1 Create the report

To create a chart that links to a report, you must first create the report that displays information related to that displayed in the chart. You will create the chart in Section 4.3.4, "Create a sales chart".

1. In the Navigator, click the **Applications** tab.

2. Click **Casebook30_Application**.

Note: If you do not see this application, contact your portal administrator. Make sure you choose the <DEMO SCHEMA> schema, where DEMO SCHEMA is the schema where the demo schema objects are installed. If you do not have access to this schema, contact your database administrator or refer to Appendix A.

3. Next to Create New, click **Report**.


4. Click **Reports From Query Wizard**.

5. In the **Name** field, enter <YourName>_REGIONAL_SALES_REPORT.

6. In the **Display Name** field, enter <YourName> Regional Sales Report.

7. Make sure **Casebook30_Application** is chosen from the Application list.

8. Click **Next**.

9. Next to the **Tables or Views** field, click  .

10. In the search field, enter <DEMO SCHEMA> .HLW_SALES_VIEW and click **Find**.

Note: The name of the table should be prefixed by the schema that was installed by your portal administrator. If you do not see the table, or if you're not sure what the schema name is, contact your portal administrator.

11. When the view displays, click it.



The Search window closes, and the view name now displays in the **Table or Views** field.

12. Click **Add**.

The view name displays under **Selected Tables/Views**.

13. Click **Next**.

14. On the Table or View Columns tab, click  to move all the columns to the Selected Columns list.

15. Use  and  to reorder the columns, so that they display like this:

SALES_VIEW.REGION

SALES_VIEW.ID

SALES_VIEW.CUSTOMER

SALES_VIEW.AMOUNT

16. Click **Next**.
17. On the Column Conditions tab, click **Next**.
18. On the Report Layout tab, make sure **Tabular** is selected.
19. Click **Next**.
20. On the Column Formatting tab, next to SALES_VIEW.ID, change the **Column Heading** to Order#.
21. Next to SALES_VIEW.ID, under Align, choose **Left**.
22. Click **Next**.
23. On the Formatting Conditions tab, make the following changes to the first row:
 - From the **Column** list: choose **NO CONDITION**.
 - From the **Row/Col** list: choose <DEMO_SCHEMA>_SALES_VIEW.AMOUNT.
 - From the **Bg Color** list: choose **Green, Pale**.
24. Click **Next**.
25. On the Display options tab, under Full Page Options, choose **Thin Border** from the **Border** list.
26. From the **Heading Background color** list, choose **Green, Pale**.
27. From the **Table Row Color(s)** list, choose **Gray, Light**.
28. Under Portlet Options, make the same changes to the Border, Heading Background, and Table Row Color.
29. Under Break Options, choose **Left Break** from the **Break Style** list.
30. From the **First Break Column** list, choose SALES_VIEW.REGION.
31. Under Row Order options, choose SALES_VIEW.REGION from the **Order By** list. Make sure **Ascending** is chosen.
32. From the first **Then By** list, choose SALES_VIEW.CUSTOMER, then make sure **Ascending** is chosen.
33. From the second **Then By** list, choose SALES_VIEW.ID, then make sure **Ascending** is chosen.
34. Click **Next**.

35. Under Customization Form Display Options, make the following changes to the first row:
 - n Select the **Value Required** check box.
 - n From the **Column Name** list, choose SALES_VIEW.REGION.
 - n In the **Prompt** field, enter REGION.
36. Under Formatting Options, clear all of the check boxes:
 - n Output Format
 - n Maximum Rows
 - n Break Columns
 - n Font Size
 - n Order By
 - n Layout format
37. Click **Finish**.
38. Click the **Access** tab.
39. Select the **Publish to Portal** check box.
40. Click **Apply**, then click **Close**. Your report is now finished.

4.3.2 Testing the sales report

Before you create the chart, you should test the sales report to make sure it works.

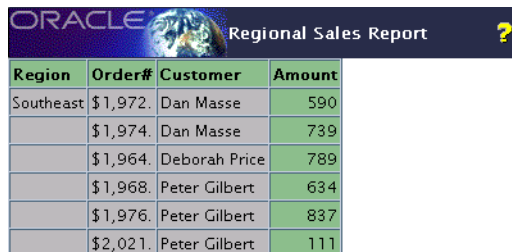
1. Display the Develop tab of the Manage Component dialog for the <YourName>_REGIONAL_SALES_REPORT. This dialog should be displaying already if you are continuing from Section 4.3.1, "Create the report".
2. Click **Run**. Notice that the component displays in a new Web browser window.

Note: You should get an error: Required field not set yet. This error occurs because you set the report's customization form to request a value.
3. Close the browser window and return to the Web browser window that currently displays the Manage Component dialog for the REGIONAL_SALES_REPORT.
4. In the Manage Component dialog, click **Customize**.
5. From the REGION drop-down list, choose =.
6. In the field, enter Southeast.

Note: This field is case-sensitive.

7. Click **Run Report**.

The report that displays should look something like this:

The screenshot shows a web interface for an Oracle report titled "Regional Sales Report". It features a table with four columns: Region, Order#, Customer, and Amount. The data is filtered by the "Southeast" region and lists seven sales entries with their respective order numbers, customer names, and amounts.

Region	Order#	Customer	Amount
Southeast	\$1,972.	Dan Masse	590
	\$1,974.	Dan Masse	739
	\$1,964.	Deborah Price	789
	\$1,968.	Peter Gilbert	634
	\$1,976.	Peter Gilbert	837
	\$2,021.	Peter Gilbert	111

4.3.3 Create a link to your sales report


In this section, you create the link that connects the chart to the report. Doing so enables you to click on the returned values in the chart to drill down to more information on that value, based on the Regional Sales report you created in Section 4.3.1, "Create the report".

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Application Tasks** and click **Go**. In the right pane, click **Building Components**, then click **Using a link between components**.

1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **Link**.
4. In the **Link Name** field, enter <YourName>_REGIONAL_SALES_REPORT_LINK.
5. From the Application field, make sure **Casebook30_Application** is chosen.
6. Click **Next**.
7. Make sure the **Oracle Portal Component** radio button is selected.
8. In the **Target Component or URL** field, enter CASEBOOK_HLW.<YourName>_REGIONAL_SALES_REPORT.
9. Click **Finish**.

4.3.4 Create a sales chart

The steps in this section help you create the chart that you will then link to the report you created in Section 4.3.1, "Create the report".

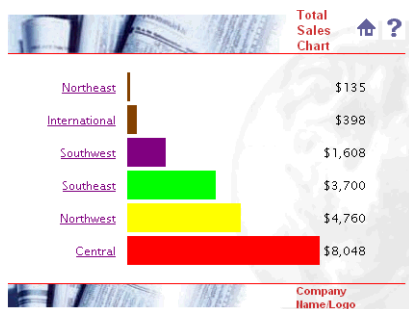
1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **Chart**.
4. Click **Chart From Query Wizard**.
5. In the **Name** field, enter <YourName>_TOTAL_SALES_CHART.
6. In the **Display Name** field, enter <YourName> Total Sales Chart.
7. From the **Application** list, make sure **Casebook30_Application** is chosen.
8. Click **Next**.
9. In the **Table/Views** field, enter <DEMO_SCHEMA> .HLW_SALES_VIEW.
Tip: If you get an error when trying to use this view, try using the Search window. If you do not see the <DEMO_SCHEMA> schema, contact your portal administrator.
10. Click **Next**.
11. On the Table or View Columns tab, from the **Label** list, choose **REGION**.
12. From the **Link** list, choose <YourName>_REGIONAL_SALES_REPORT_LINK (the link you just created in Section 4.3.3, "Create a link to your sales report").
13. Next to the **Link** list, click .
14. From the **Condition** list, choose =.
15. From the first **Column Name** list, choose **REGION**.
16. Click **OK**.
17. From the **Value** list, choose **AMOUNT**.
18. From the **Group Function** list, choose **SUM**.
19. Click **Next**.
20. On the **Column Conditions** tab, click **Next**.
21. On the **Display Options** tab, under Common Options, choose **ORDER BY VALUE** from the **Order By** list.

22. Under Full Page Options, in the **Value Format Mask** field, enter \$ in front of the numerals 999,999,999,999.
23. Click **Finish**.
24. In the Manage Component dialog, click **Access**.
25. Select the **Publish to Portal** check box.
26. Click **Apply**, then **Close**. Your chart is now finished.

4.3.5 Test the chart with links

Now that you've create the report, the link, and the chart, try testing the chart to make sure the links display the corresponding data in the report.

1. In the Manage Component dialog, on the Develop tab, click **Run**. You should see something like this:




2. Click **Northeast**. The following should display:

A screenshot of an Oracle report titled "Regional Sales Report" with an Oracle logo and a question mark icon. The report displays a table with four columns: Region, Order#, Customer, and Amount. The data shown is for the Northeast region, specifically for two orders.



Region	Order#	Customer	Amount
Northeast	\$1,966.	Hilary Walters	80
	\$1,980.	Hilary Walters	55

4.3.6 Display the chart as a portlet

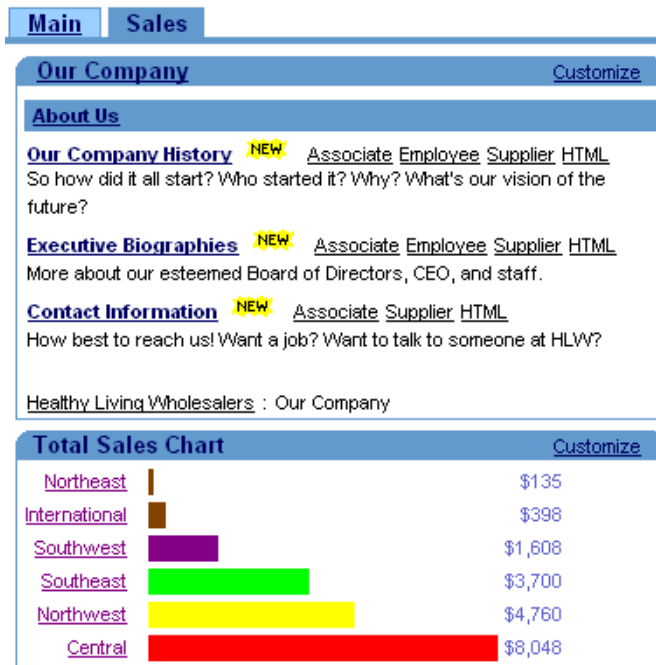
After you've created the chart with links, you can now display it as a portlet. When you click the links in the chart, the corresponding field in the report will display.

1. In the Navigator, click the **Pages** tab.
2. Navigate to **<YourName> Inside Healthy Living**.
3. Click **Edit**.
4. Make sure the **Sales** tab is highlighted.
5. Click  to add a new region.

The Sales page should now be divided into two regions, with the folder portlet you added in Section 4.2, "Add an existing folder as a portlet" displaying in the left region.

6. On the **Portlets** tab, in the left pane, click  .
7. Scroll down the **Providers** list until you find **Casebook30_Application**.
8. Under Casebook30_Application, find the chart called **<YourName> Total Sales Chart**.
9. Next to the chart, click  .
10. Click **OK**.
11. Click **Close**.

The chart displays in your portal.



4.4 Create a multi-tabbed region containing two charts

The steps in this section help you create two different charts: one that displays the sales budget for the year, and one that displays the top revenue for each product. You will then display the two charts in a single portlet, along with the chart you created in Section 4.3, "Create a sales chart portlet that links to a report". Doing so enables the salespeople to easily reference each chart, as well as organize their information. You can also use multiple tabs in a region to easily display different views of the same data in the same region.



4.4.1 Use the available charts

For your convenience, we've created the two charts you will add to a single portlet. To access these charts, you must have Edit or higher schema access privileges. If you do not have these privileges (or do not know), contact your portal administrator.

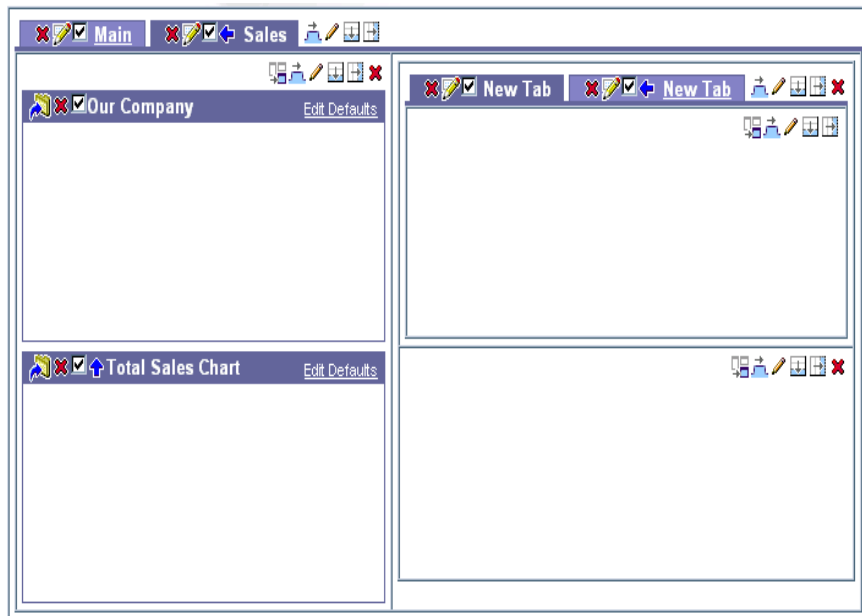
Note: For the purposes of this exercise, you can use any charts that you have access to.


4.4.2 Add the charts to the same portlet




In this section, you will create a multi-tabbed portlet, then add a chart to each of the tabs.

1. In the Navigator, click the **Pages** tab.
2. Click **My Pages**.
3. Next to the Inside Healthy Living Page, click **Edit**.
4. On the Portlets tab, make sure the **Sales** tab is highlighted.
5. In the right-hand column, click  to add a row to your new column.
6. In the top region of the right-hand column, click  to add a tab to the region.
7. Repeat step 7 so that two tabs display in the same region.

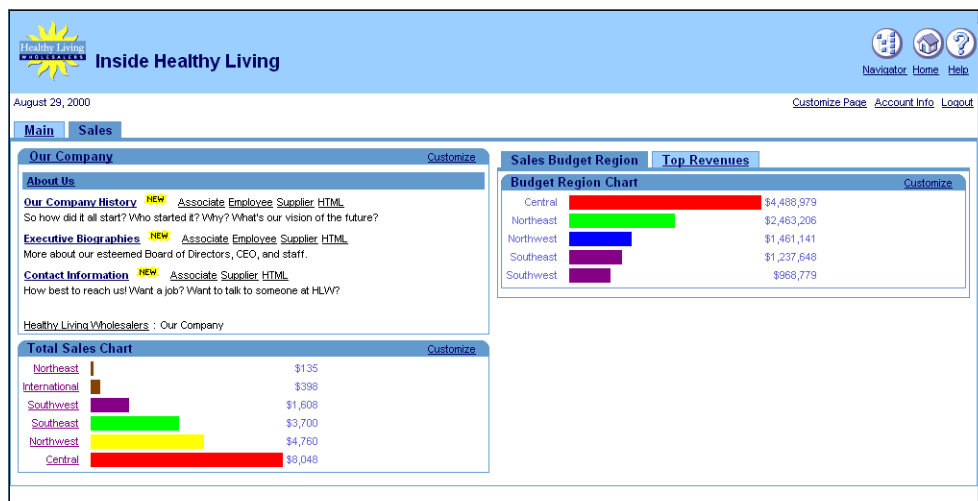
The layout should now look something like this:



8. Next to the first new tab, click .
9. In the **Display Name** field, enter Sales Budget by Region.
10. Click **Apply**.

11. Click **Close**.
12. Follow steps 10 - 12 for the second tab and name it **Top Revenues**.
13. Click the **Sales Budget by Region** tab.
14. Click .
15. In the **View By** list, make sure **Provider** is chosen.
16. Scroll down until you find the heading **Casebook Application**.
17. Next to the chart named **Budget Region Chart**, click .
18. Next to the chart named **Top Revenue Chart**, click .
19. Click **OK**.
20. Click **Close**.

The tabbed portlet displays on the Sales tab of the portal. The resulting page should look something like this:



4.5 Create a multi-tabbed region containing various components

The steps in this section help you create a chart, a report, and a search form that lets you access all sales commission information. You will then display these three components in a

single region. Doing so will make it easy for salespeople to view their department's commission information in three different ways.

4.5.1 Create a sales commission chart

In this section, you use the Charts from SQL Query Wizard. Creating a chart based on a SQL query is powerful because you can create your own SQL statement. If you want to add a link to this chart, you can also add the link's SQL statement to the SQL statement in the chart.

1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **Chart**.
4. Click **Charts From SQL Query**.
5. In the **Name** field, enter `<YourName>_COMM_REG_CHART`.
6. In the **Display Name** field, enter `<YourName> Sales Commission Chart`.
7. Make sure **Casebook30_Application** is chosen from the **Application** list.
8. Click **Next**.
9. In the **SQL Query** field, enter something like this:

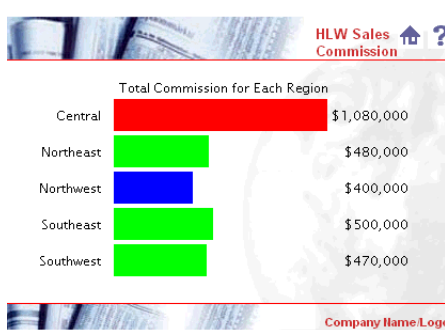
```
select
    null      the_link,
    r.region_name Region,
    sum(e.commission) Commission
from <DEMO_SCHEMA>.hlw_staff e, <DEMO_SCHEMA>.hlw_regions r
where r.region_id = e.region_id
group by region_name
order by 1
```

Note: Do not copy and paste the code directly from this text. You must change the name of the table according to the name of the schema where the demo schema objects are installed.

10. Click **Next**.

Note: If you receive an error at this point, you may have entered the SQL query incorrectly. Verify that your SQL statement is accurate. If the statement is correct and you still receive an error, check with your portal administrator to verify that you have access to the `<DEMO_SCHEMA>` schema.

11. In the **Value Format Mask** field, enter \$ in front of the 999,999,999,999.
12. Click **Next**.
13. On the Customization Form Display Options tab, click **Next**.
14. In the **Header Text** field, enter Total Commission for Each Region.
15. Click **Finish**.
16. In the Manage Component dialog, click **Access**.
17. Select the **Publish to Portal** check box.
18. Click **Apply**.
19. Click **Develop**.
20. Click **Run** to test the new chart. The chart should look something like this:



4.5.2 Create a sales commission report

In this section, you use the Reports from Query Wizard to create a report that will display on one of the tabs in the multi-tabbed portlet.

1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **Report**.
4. Click **Reports From Query Wizard**.
5. In the **Name** field, enter <YourName>_COMMISSION_REPORT.
6. In the **Display Name** field, enter <YourName> Sales Commission Report.

7. Make sure **Casebook30_Application** is chosen from the **Application** list.

8. Click **Next**.

9. Add the two following tables:

<DEMO_SCHEMA>.HLW_STAFF

<DEMO_SCHEMA>.HLW_REGIONS

Note: DEMO_SCHEMA is the name of the schema in which the demo schema objects are installed. To add the tables, enter each table name in the field, then click **Add**. Once you've added the tables, the table names display below the field.

10. Click **Next**.

11. On the Join Conditions tab, you should see a default join display:

<DEMO_SCHEMA>_HLW_REGIONS.REGION_ID = <DEMO_SCHEMA>_HLW_STAFF.REGION_ID

If this default conditions displays, leave it and go to the next step. If this default condition does not display, create the join by choosing the appropriate columns from the lists.

12. Click **Next**.

13. Select the following columns by clicking on each in the **Columns** list, then clicking **>** to move the column to the **Selected Columns** list:

HLW_REGIONS.REGION_NAME

HLW_STAFF.EMPLOYEE_ID

HLW_STAFF.NAME

HLW_STAFF.COMMISSION

14. Click **Next**.

15. In the first row, from the **Column Name** list, choose **HLW_STAFF.COMMISSION**.

16. From the **Condition** list, choose **not null**.

17. Click **Next**.

18. Make sure **Tabular Layout** is chosen.

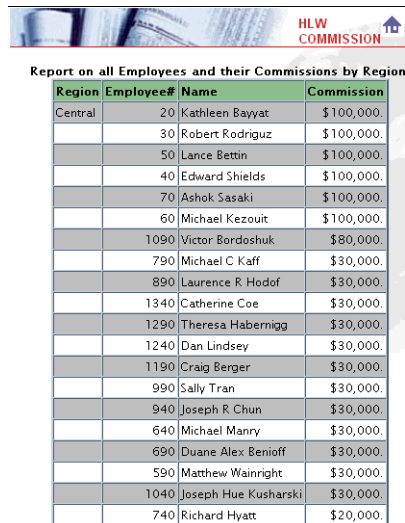
19. Click **Next**.

20. Under **Column Heading**, change Region Name to **Region**.

21. Change Employee Id to **Employee#**.

22. Next to EMPLOYEES.COMMISSION, under **Format Mask**, enter \$999,999.
23. Click **Next**.
24. On the Display Options tab, click **Next**.
25. Under Full Page Options, choose **Thick Border** from the **Border** list.
26. From the **Heading Background Color** list, choose **Green, Pale**.
27. Under Portlet Options, change the **Border** to **Thick Border** and the **Heading Background Color** to **Green, Pale**.
28. Under Break Options, make sure **Left Break** is chosen.
29. From the **First Break Column** list, choose **HLW_REGIONS.REGION_NAME**.
30. Under Row Order Options, choose **HLW_REGIONS.REGION_NAME** from the **Order By** list, then choose **Ascending**.
31. In the next row, choose **HLW_STAFF.COMMISSION** from the **Order By** list, then choose **Descending**.
32. Click **Next**.
33. On the Customization Form Display Options tab, click **Next**.
34. On the Report and Customization Form Text tab, in the **Header Text** field, enter:

```
<H4>Report on all Employees and their Commissions by Region</H4>.
```
35. Click **Finish**.
36. In the Manage Component dialog, click **Access**.
37. Select the **Publish to Portal** check box.
38. Click **Apply**.
39. Click **Develop**.
40. Click **Run** to test the report. The report should look something like this:



Report on all Employees and their Commissions by Region

Region	Employee#	Name	Commission
Central	20	Kathleen Bayyat	\$100,000.
	30	Robert Rodriguz	\$100,000.
	50	Lance Bettin	\$100,000.
	40	Edward Shields	\$100,000.
	70	Ashok Sasaki	\$100,000.
	60	Michael Kezouit	\$100,000.
	1090	Victor Bordoshuk	\$80,000.
	790	Michael C Kaff	\$30,000.
	890	Laurence R Hodof	\$30,000.
	1340	Catherine Coe	\$30,000.
	1290	Theresa Habernigg	\$30,000.
	1240	Dan Lindsey	\$30,000.
	1190	Craig Berger	\$30,000.
	990	Sally Tran	\$30,000.
	940	Joseph R Chun	\$30,000.
	640	Michael Manry	\$30,000.
	690	Duane Alex Benioff	\$30,000.
	590	Matthew Wainright	\$30,000.
	1040	Joseph Hue Kusharski	\$30,000.
	740	Richard Hyatt	\$20,000.

4.5.3 Create four LOVs for the commission search form

In this section you will create four lists of values (LOVs) that you will then use in Section 4.5.4, "Create a commission search form". One of these LOVs will depend on the items you choose in another LOV.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Application Tasks** and click **Go**. In the right pane, click **Building Components**, then click **Building Lists of Values**.

4.5.3.1 Create JOB_LIST dynamic list of values

1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **List of Values**.
4. Click **Dynamic List of Values**.
5. Make sure **Casebook30_Application** is chosen in the Owner drop-down list.
6. In the **Name** field, enter <YourName>_JOB_LIST.
7. From the **Default format** list, choose **Combo box**.
8. From the **Show Null Value** list, choose **Yes**.

9. In the **SQL Query** field, enter something like this:

```
select distinct job
from <DEMO_SCHEMA>.hlw_staff
order by job
```

Note: Do not directly copy and paste the SQL from this text. You must replace the schema name with the name of the schema where the demo schema objects are installed.

10. Click **OK**.

4.5.3.2 Create REGION_LOV dynamic list of values

1. Follow steps 1 through 5 from Section 4.5.3.1, "Create JOB_LIST dynamic list of values".
2. In the **Name** field, enter <YourName>_REGION_LOV.
3. From the **Default format** list, choose **Combo box**.
4. From the **Show Null Value** list, choose **Yes**.
5. In the **SQL Query** field, enter something like this:

```
select region_name, region_id
from <DEMO_SCHEMA>.hlw_regions
```

Note: Do not directly copy and paste the SQL from this text. You must replace the schema name with the name of the schema where the demo schema objects are installed.

6. Click **OK**.

4.5.3.3 Create DEPARTMENT_LOV static list of values

This LOV will control the items that display in the LOV you will create in Section 4.5.3.4, "Create EMP_LOV dynamic list of values with a bind variable". When you choose a department number from this LOV, the employees of that department will display in the Employee LOV.

1. Follow steps 1 through 2 from Section 4.5.3.1, "Create JOB_LIST dynamic list of values".
2. Click **Static List of Values**.
3. Make sure **Casebook30_Application** is chosen in the Owner drop-down list.

4. In the **Name** field, enter <YourName>_DEPARTMENT_LOV.
5. From the **Default format** list, choose **Combo box**.
6. From the **Show Null Value** list, choose **Yes**.
7. In the first row, in the **Display Value** field, enter Head Office.
8. In the **Return Value** field, enter 10.
9. In the **Display Order** field, enter 1.
10. Add the following values:

Table 4–1 Department Static List of Values

Manufacturing	20	2
Development	30	3
Sales	40	4
Shipping	50	5
Operations	60	6
Human Resources	70	7

11. Click **OK**.

4.5.3.4 Create EMP_LOV dynamic list of values with a bind variable

This LOV will change depending on the item chosen from the Department List of Values. When you choose a department number from the DEPARTMENT_LOV, the employees of that department will display in this LOV. To create an LOV that depends on another, you must use a bind variable in its SQL query.

1. Follow steps 1 through 5 from Section 4.5.3.1, "Create JOB_LIST dynamic list of values".
2. In the **Name** field, enter <YourName>_EMP_LOV.
3. From the **Default format** list, choose **Combo box**.
4. From the **Show Null Value** list, choose **Yes**.
5. In the **SQL Query** field, enter something like this:

```
select NAME, employee_id
from <DEMO_SCHEMA>.hlw_staff
where department_no = :deptno
order by name
```

Note: Do not directly copy and paste the SQL from this text. You must replace the schema name with the name of the schema where the demo schema objects are installed.

6. Click **OK**.


4.5.4 Create a commission search form


Before you can complete this section, you must create the lists of values (see Section 4.5.3, "Create four LOVs for the commission search form"). If you do not want to create the lists of values first, you can always edit this form later.

1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **Form**.
4. Click **Forms based on table or view**.
5. In the **Name** field, enter <YourName>_COMM_SEARCH_FORM.
6. In the **Display Name** field, enter <YourName> Sales Commission Lookup.
7. Make sure **Casebook30_Application** is chosen from the **Application** list.
8. Click **Next**.
9. Add the <DEMO_SCHEMA>.HLW_STAFF table.

Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed. To add a table, enter the table name in the Table or View field.


10. Click **Next**.
11. Make sure **Tabular** is selected.
12. Click **Next**.
13. In the left pane, display the items below FORM.

Note: If you do not see any items listed below FORM in the left pane, click  next to FORM.

14. Next to each of the following items, click  to remove the item from the form, then click **OK** to accept the dialog:

- n INSERT_TOP
- n UPDATE_TOP
- n DELETE_TOP
- n EMPLOYEE_GENDER
- n MARITAL_STATUS
- n USER_ID
- n EMAIL_ID
- n HIRE_DATE
- n ANNUAL_SALARY
- n MANAGER_ID
- n BUDGET
- n VACATION_DAYS
- n EMPLOYEE_ADDRESS
- n EMPLOYEE_CITY
- n EMPLOYEE_ZIP
- n EMPLOYEE_STATE
- n EMPLOYEE_PHONE
- n INSERT_BOTTOM
- n UPDATE_BOTTOM
- n DELETE_BOTTOM
- n QUERY_BOTTOM
- n RESET_BOTTOM

15. In the left pane, click **FORM**. The form's properties display in the right pane.
16. From the **Box Background Image** list, choose **Dude**.
17. From the **Box Border** list, choose **Thick Border**.
18. Clear the **Show Timing** check box.

19. Next to Order By, choose **EMPLOYEE_ID** from the list, and make sure **Ascending** is chosen.
20. In the left pane, next to **DEPARTMENT_NO**, click  to move DEPARTMENT_NO above EMPLOYEE_ID.
21. In the left pane, click **QUERY_TOP**.
22. In the **Label** field, enter Search.
23. In the left pane, click **DEPARTMENT_NO**.
24. From the **Item Type** list, choose **Combobox**.
25. In the **Label** field, enter Department Number.
26. From the **List of Values** list, choose <YourName>_DEPARTMENT_LOV.
Tip: Choose the LOV you created in Section 4.5.3.3, "Create DEPARTMENT_LOV static list of values".
27. In the left pane, click **EMPLOYEE_ID**.
28. From the **Item Type** list, choose **Combobox**.
29. In the **Label** field, enter Employee.
30. From the **Font Color** list, choose **Black**.
31. From the **List of Values** list, choose <YourName>_EMP_LOV.
Tip: Choose the LOV you created in Section 4.5.3.4, "Create EMP_LOV dynamic list of values with a bind variable".
32. From the **Set binding(s) for the bind variable(s) defined in the lov deptno** list, choose **DEPARTMENT_NO**.
33. In the left pane, click **NAME**.
34. In the **Input Width** field, enter 30.
35. In the left pane, click **JOB**.
36. From the **Item type** list, choose **ComboBox**.
37. In the **Label** field, enter Position.
38. From the **LOV** list, choose <YourName>_JOB_LIST
39. In the left pane, click **REGION_ID**.
40. From the **Item type** list, choose **ComboBox**.

41. In the **Label** field, enter **Region**.
42. From the **LOV** list, choose **<YourName>_REGION_LOV**.
43. Click **Next**.
44. In the **Header Text** field, enter **Search for Employee Commission**.
45. Click **Finish**.
46. In the Manage Component dialog, click **Access**.
47. Select the **Publish to Portal** check box.
48. Click **Apply**.
49. Click **Develop**.
50. Click **Run** to test the form. The form should look something like this:

Note: You should be able to choose a job title, a department name, and a region name from each of the three drop-down lists. These drop-down lists are based on the three LOVs you created in Section 4.5.3, "Create four LOVs for the commission search form".

4.5.5 Add the chart, report, and form to the same portlet

In this section, you will add all the components you created to the same portlet.

1. In the Navigator, click the **Pages** tab.
2. Click **My Pages**.
3. Next to **<YourName> Inside Healthy Living**, click **Edit**.
4. Follow the steps in Section 4.4.2, "Add the charts to the same portlet" and create three tabs with the following names:

Commission Chart

Commission Report

Commission Lookup

Note: Remember to click **Apply** when you change the name of the tab.

5. Add the chart, **<YourName> Sales Commission Chart**, to the Commission Chart tab.
6. Add the report, **<YourName> Sales Commission Report**, to the Commission Report tab.
7. Add the form, **<YourName> Sales Commission Lookup**, to the Commission Lookup tab.
8. Close the Customize Page dialog. The portlet should look something like this (the tab containing the commission form is currently active):

The screenshot shows a web application interface with three tabs: 'Commission Chart', 'Commission Report', and 'Commission Lookup'. The 'Commission Lookup' tab is currently selected and active. Below the tabs is a portlet titled 'Search for Commission' with a 'Customize' and 'Help' link. The portlet contains a search form with the following fields: 'Employee ID', 'Name', 'Position' (a dropdown menu), 'Commission', 'Department' (a dropdown menu, highlighted in red), 'Region' (a dropdown menu), and 'Ccid'. There are 'Search' and 'Reset' buttons at the top of the form area.

4.6 Create a customer entry form

The steps in this section help you create an entry form in which salespeople can add, update, and remove customer information from the database. You can create forms in Oracle Portal that display as portlets, to enable users to easily update and access information in the database.

4.6.1 Create a database sequence

In this section, you create a database sequence that the form uses to assign each customer record a unique ID.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Administration Tasks** and click **Go**. In the right pane, click **Building Database Objects**.

1. In the Navigator, click **Database Objects**.

Note: You must have **MANAGE ALL SCHEMAS** global privilege or **Manage** privileges on the <DEMO_SCHEMA> schema to create database objects.

2. Under Name, click <DEMO_SCHEMA>.
3. Next to Create New, click **Sequence**.
4. In the **Sequence Name** field, enter <YourName>_NEW_CUSTOMER.
Tip: It's always a good idea to give your database objects (or any component) a meaningful name, particularly when you share your applications with other users.
5. In the **Schema** list, make sure <DEMO_SCHEMA> is chosen.
6. Click **Next**.
7. In the **Increment By** field, enter 1.
8. In the **Start With** field, enter 1000.

Note: Setting the two fields with these values causes the customer entries in the database to begin with the number 1000. Every following record will increment by 1, such that the second customer record will have the ID of 1001.

9. Click **Finish**.
10. In the Navigator, find the new database sequence (<YourName>_NEW_CUSTOMER) on the Database Objects tab.
11. Next to the sequence, click **Grant Access**.
12. In the **User/Role** field, enter PUBLIC.
13. Click **SELECT** in the list.
14. Click **Add to List**.
15. Click **Close**.



You have now created a database sequence number that can be used by the form you create in the next section.

4.6.2 Create a customer entry form

In this section, you will create a form where salespeople within the Healthy Living company can enter new customers, or update existing customers' information.

1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **Form**.
4. Click **Forms based on tables or views**.
5. In the **Name** field, enter <YourName>_CUSTOMER_ENTRY_FORM.
6. In the **Display Name** field, enter <YourName> Customer Entry Form.
7. In the Application field, make sure **Casebook30_Application** is chosen.
8. Click **Next**.
9. In the **Table or View** field, enter <DEMO_SCHEMA> .HLW_CUSTOMERS.

Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed.

10. Click **Next**.
11. Select the **Tabular** radio button.
12. Click **Next**.
13. In the left pane, under FORM, you should see a list of table columns. If you do not see anything listed under FORM, click  , next to FORM.
14. Next to each of the following items, click  to remove the item from the form, then click **OK** to accept the dialog:

- ☐ PASSWORD
- ☐ GENDER
- ☐ AGE
- ☐ INCOME
- ☐ CURRENCY
- ☐ CREDIT_CARD_OWNER_NAME
- ☐ CREDIT_CARD_TYPE
- ☐ CREDIT_CARD_NUMBER

- n CREDIT_CARD EXPIRATION_DATE
- n ADDRESS_LINE2
- n INSERT_BOTTOM
- n UPDATE_BOTTOM
- n DELETE_BOTTOM
- n QUERY_BOTTOM
- n RESET_BOTTOM

15. At the top of the left pane, click **FORM** to display the Form-Level properties for the form.
16. In the right pane, under Form Level Options, choose **World** from the **Box Background Image** list.
17. From the **Box Border** list, choose **Thick Border**.
18. Clear the **Show Timing** check box.
19. From the **Order by** list, choose **ID**, then make sure **Ascending** is chosen.
20. In the left pane, click **ID** to display the Item Level Options for ID.
21. In the right pane, in the **Label** field, enter `Customer ID`.
22. Under Validation Options, in the **Default Value** field, enter `#<DEMO_SCHEMA>.<YourName>_NEW_CUSTOMER.nextval`.

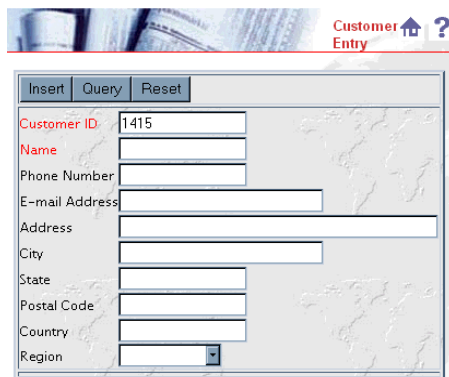
Note: DEMO_SCHEMA is the name of the schema where the schema objects are installed. Also, entering a default value tells this item to base the customer ID number on the database sequence you created in Section 4.6.1, "Create a database sequence".
23. In the left pane, click **PHONE_NUMBER** to display its properties.
24. In the right pane, in the **Label** field, enter `Phone Number`.
25. In the left pane, click **EMAIL_ADDRESS** to display its properties.
26. In the right pane, in the **Label** field, enter `E-mail Address`.
27. In the **Input Width** field, enter 25.
28. In the left pane, click **ADDRESS_LINE1** to display its properties.
29. In the right pane, in the **Label** field, enter `Address`.
30. In the **Input Width** field, enter 40.

31. In the left pane, click **CITY**.
32. In the right pane, in the **Input Width** field, enter 25.
33. In the left pane, click **POSTAL_CODE**.
34. In the right pane, in the **Label** field, enter Postal Code.
35. In the left pane, click **REGION_ID**.
36. In the right pane, choose **ComboBox** from the **Item type** list.
37. In the **Label** field, enter Region.
38. In the **List of Values** field, choose <YourName>_REGION_LOV.

Note: If you did not complete Section 4.5.3.2, "Create REGION_LOV dynamic list of values", this LOV will not display. You can save this form, create the REGION_LOV by following the steps in Section 4.5.3.2, "Create REGION_LOV dynamic list of values", then return to this tab of the form to complete the Region field.

39. Click **Finish**.

Note: You can always go back and change the **Input Width** for each of the fields, depending on the data you are using.





4.7 Create a product report

The steps in this section help you create a report that lists all the products contained in the database. In Section 4.8, "Create a menu containing links", you create a menu portlet that enables users to access this report via a link. You can skip this section if you've already created a report.


You will also use the customization form to allow users to customize the report. The details you specify on this form display when users click the Customize link in the portlet. This way, you can control the default data that displays on the portal page, but users can narrow the report results using the form you create.

1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **Report**.
4. Click **Reports From Query Wizard**.
5. In the **Report Name** field, enter <YourName>_PRODUCT_LISTINGS.
6. In the **Display Name** field, enter <YourName> Product Listings.
7. In the **Application** field, make sure **Casebook30_Application** is chosen.
8. Click **Next**.
9. In the **Table or Views** field, enter <DEMO_SCHEMA> .HLW_PRODUCTS.

Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed.

10. Click **Add**. The table displays in the **Selected Tables/Views** list.
11. Click **Next**.
12. In the **Columns** list, select the following columns (in Windows, you can press CTRL and click to select multiple columns):
 - HLW_PRODUCTS.ID
 - HLW_PRODUCTS.NAME
 - HLW_PRODUCTS.DESCRPTION
 - HLW_PRODUCTS.PRICE
13. Click  to move all the selected columns to the **Selected Columns** list.
14. In the Selected Columns list, click **PRODUCTS.PRICE**.
15. Click  to move this column to the end of the list.
16. Click **Next**.
17. On the Column Conditions tab, click **Next**.
18. On the Report Layout tab, make sure the **Tabular Layout** radio button is selected, then click **Next**.

19. Next to HLW_PRODUCTS.ID, change the **Column Heading** from Id to Product #.
20. Next to HLW_PRODUCTS.ID, from the **Align** list, choose **Left**.
21. Next to HLW_PRODUCTS.DESCRPTION, from the **Width Type** list, choose **Char**.
22. Next to HLW_PRODUCTS.DESCRPTION, in the **Width** field, enter 90.
23. Next to HLW_PRODUCTS.PRICE, in the **Format Mask** field, enter \$999,999.99.
24. Click **Next**.
25. On the Formatting Conditions tab, click **Next**.
26. Under Full Page Options, choose **Thin Border** from the Border drop-down list,.
27. From the **Heading Background Color** list, choose **Green, Pale**.
28. Under Portlet Options, change the border to **Thin Border** and the **Heading Background Color** to **Green, Pale**.
29. Under Row Order Options, choose **PRODUCTS.ID** from the **Order By** list, then choose **Ascending**.
30. Click **Next**.
31. On the Customization Form Display Options tab, click **Next**.
32. In the **Header Text** field, enter <CENTER>.
33. In the **Footer Text** field, enter </CENTER>.
34. Click **Finish**.
35. In the Manage Components dialog, click **Access**.
36. Select the **Publish to Portal** check box.
37. Click **Apply**.
38. Click **Develop**.
39. Click **Run**. The report that displays should look something like this:



Product #	Name	Description	Price
323	Knoggin Ski Helmet	Looking for the best protection for your cranial region? Look no further! This is the ultimate. Cheap copies have been made for others, but their heads have been cracked like eggshells on the side of many mountains.	\$59.00
326	Killa Snowboard	This snowboard is da bomb. Only available in goofy.	\$103.00
370	Jansen Baby Powder	For the softest smoothest skin imaginable, Jansen's Baby Powder is gentle enough for daily use for you and baby.	\$5.00
410	Krazo Vitamin D	Exclusively offered by Krazo. Available nowhere else but here!	\$43.00
411	"Super" Vitamin K	Helps cells grow normally, stabilizes bone tissue.	\$21.95
414	TriChem Vitamin B3	Following upon their success with Vitamin B2, comes their successor – Vitamin B3! Now nearly twice as effective as B2 in most clinical trials.	\$5.99
423	Beyodi Skin Cleanser	Gently and thoroughly dissolves dirt, oil, makeup, even mascara, while rinsing clean to give you the freshest complexion ever.	\$8.99

4.8 Create a menu containing links

In this section, you will create a menu that provides access to various components you've created, such as the Customer Entry Form. Creating menus provides a way of organizing components, folders, and HTML pages in Oracle Portal that users want to easily access from the portal.

4.8.1 Create two links for your menu

Before you create a menu, you must create the links that the menu will hold. When the user clicks these links, the corresponding component, page, or item will display.

1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **Link**.
4. In the **Link Name** field, enter <YourName>_PRODUCT_REPORT.
5. Click **Next**.
6. Select the **Oracle Portal Component** radio button.
7. In the **Target Component or URL** field, enter Casebook30_Application.PRODUCT_LISTINGS.



Note: This field is case-sensitive.


8. Click **Finish**.
9. In the Manage Component dialog, click **Access**.
10. Make sure the **Inherit Privileges from Application** check box is selected.
11. Click **Close**.
12. Follow steps 3 through 11 to create a second link with the following details:
 - **Name** = NEW_CUSTOMER
 - **Target Component or URL** = Casebook30_Application.CUSTOMER_ENTRY_FORM



After you've clicked **Close**, your links should now display in the Navigator.

4.8.2 Create a menu

In this section, you will create two menu items that will link to the Product Report and Customer Entry Form you created in this chapter. These items rely on the links you created in Section 4.8.1, "Create two links for your menu".

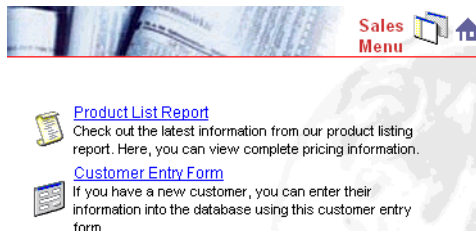
1. In the Navigator, click the **Applications** tab.
2. Click **Casebook30_Application**.
3. Next to Create New, click **Menu**.
4. In the **Menu Name** field, enter <YourName>_SALES_MENU.
5. In the **Display Name** field, enter <YourName> Sales Menu.
6. In the Application field, make sure **Casebook30_Application** is chosen.
7. Click **Next**.
8. On the Menu Items and Submenus tab, make sure **My Menu** is highlighted in the left pane.
9. In the **Name** field, enter <YourName> Sales Menu.
10. Next to **My Menu** in the left pane, click .
- Note:** The menu name updates when you click .
11. In the pop-up menu, choose **Add menu item**.
12. Make sure **MENU_ITEM_1** is selected in the left pane.

13. In the **Name** field, enter Product List Report.
14. In the **Bullet Icon** field, enter sql.gif.
15. Next to the **Link** field, click .
16. In the Search window, find <DEMO_SCHEMA>.<YourName>_PRODUCT_LISTINGS_SHOW (the first link you created in Section 4.8.1, "Create two links for your menu").
17. In the **Description** field, enter:

Check out the latest information from our product listing report. Here, you can view complete pricing information.
19. From the **Font Face** list, choose **Arial**.
20. In the left pane, next to Sales Menu, click , then choose **Add menu item**.
21. In the **Name** field, enter Customer Entry Form.
22. In the **Bullet Icon** field, enter sql.gif.
23. In the **Link** field, click .
24. In the Search window, click <YourName>_CUSTOMER_ENTRY_FORM_SHOW (the second link you created in Section 4.8.1, "Create two links for your menu"). The link should display in the Link field.
25. In the **Description** field, enter:



If you have a new customer, you can enter his information into the database using this customer entry form.
26. From the **Font Face** list, choose **Arial**.
27. Click **Finish**.
28. In the Manage Component dialog, click **Access**.
29. Select the **Publish to Portal** check box.
30. Clear the **Inherit Privileges from Application** check box.
31. Click **Apply**.
32. Click **Develop**.
33. Click **Edit**.
34. Make sure **Sales Menu** is selected in the left pane.

35. In the right pane, from the **Security** list, click **AUTHENTICATED_USERS (GROUP)**.
36. Under **Common Options** (you may have to scroll down the right pane to find this heading), clear the following check boxes:
 - ☐ **Show Timing**
 - ☐ **Show Find Option Button**
 - ☐ **Log Activity**
37. In the left pane, click **Product List Report**.
38. Follow steps 34 and 35 for the Product List Report.
39. In the left pane, click **Customer Entry Form**.
40. Follow steps 34 and 35 for the Customer Entry Form.
41. Click **OK**.
42. In the Manage Component dialog, click **Run**. The menu displays like this:

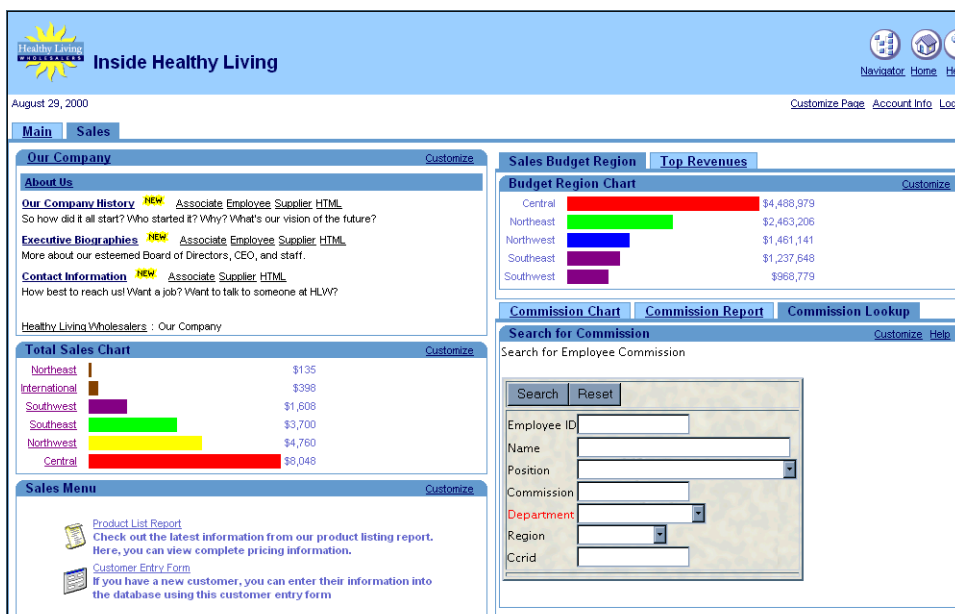


Try clicking the links to make sure the Product Report and Customer Entry Form both display.

4.8.3 Add the menu as a portlet

1. In the Navigator, click the **Pages** tab.
2. Navigate to <YourName> Inside Healthy Living.
3. Click **Edit**.
4. Make sure the **Sales** tab is highlighted.
5. In the left region, under the <YourName> TOTAL SALES chart, click .
6. In the Portlet Providers list, find the Sales Menu and click  to add it.

7. Click **OK**.
8. Display the page in your browser. The final portal should look something like this:



4.9 Summary

Congratulations! You have created an intranet portal specific to a line of business, and added different types of portlets to it. You now know how to:

- n Create a page.
- n Add a folder as a portlet.
- n Create a report using the Report Based on a SQL Query.
- n Create a chart containing links to a report
- n Create a multi-tabbed portlet containing three charts
- n Create a multi-tabbed portlet containing three views of the same data (report, chart, and form)
- n Create a customer entry form containing:
 - n static list of values

- dynamic list of values
- a dynamic list of values that depends on a static list of values
- Create a report using the Report from Query Wizard.
- Create a menu containing links
- Create a bind variable list of values

For more information about creating a portal and using portlets, see the *Oracle Portal Online Help*.

Creating a Human Resources portal with security

This chapter describes how to use Oracle Portal to build a portal for a Human Resources department in a fictitious company (Healthy Living Wholesalers), and to customize the pages of the portal for different types of users. Here, you will create pages and components, then implement different levels of security and access privileges these objects. If you are unfamiliar with creating basic pages and components, try building the samples in Chapter 3, "Creating an intranet portal and adding simple portlets" and Chapter 4, "Creating a line of business page for your intranet portal".

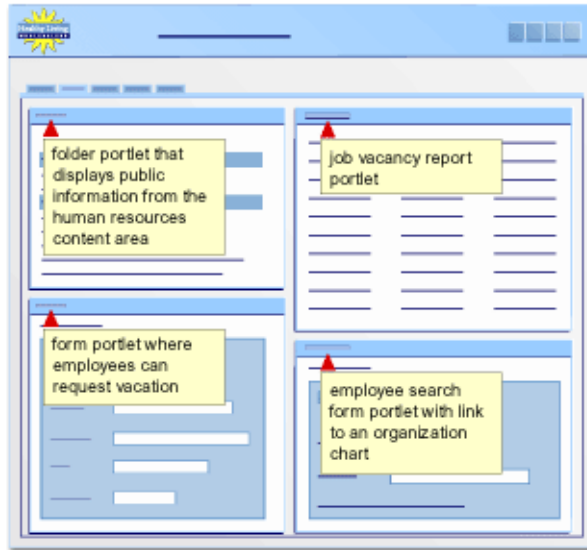
In this chapter, you will create three pages for the Human Resources portal, one for all employees (Human Resources), one just for Healthy Living managers (Managers), and one just for Human Resources representatives (HR Reps). You will use the built-in security features to display a different page for each of three types of users: public (all employees), Managers (all company managers), and HR Representatives (all Healthy Living Human Resources representatives). For more information on planning a page, see Chapter 2, "Planning Your Web Content". For information on creating the types of users, contact your portal administrator.

You will use information from an existing content area to enable users to access existing department information via the portal. You can either use the content area we've provided, called the Inside Healthy Living Repository, or create a new one by following the steps in Chapter 6, "Creating an information repository".

You will combine simple components to create complex applications that provide each type of user with a customized workplace. For example, you will create forms, a report, and a calendar that display related information. On the Human Resources tab, an employee can use a public form to submit a vacation request, a report on the Managers' tab displays how much vacation time she has left, as well as a calendar displays the duration of the requested vacation on the manager's page. A link on the calendar takes the manager to an approval form, where she can accept or reject the vacation request.

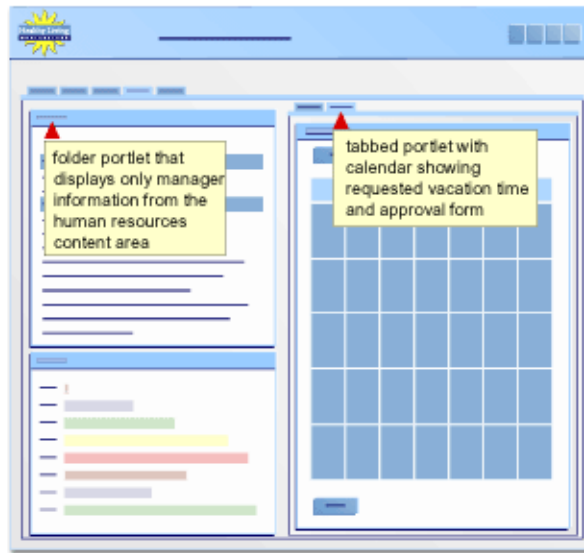
The following figure illustrates the various parts of the Inside Healthy Living portal, which contains the three tabs of the Human Resources portal.

- The **Human Resources** page, which contains the public folder of the Human Resources content area (see Chapter 6, "Creating an information repository", a general job vacancy report, a vacation request form, and an employee search form) will look something like the following:



- The **Managers** page, which contains the Managers' folder of the Human Resources content area, a salary chart, and a tabbed portlet that displays a calendar showing

requested vacation time and a form to approve the requests) will look something like the following:



- The **HR Representatives** page, which contains the HR Representatives' folder of the Human Resources content area, a staff vacation balance report, and a tabbed portlet that

displays the job vacancy report and a form to post or close vacancies) will look something like the following:



To build the Inside Healthy Living Human Resources portal, you will use the Customize Page wizard to add tabs to an existing page (or the Create Page wizard to create a page if you haven't completed Chapter 3 or 4), the Create Hierarchy wizard to create a company organization chart, and the Create Forms and Create Reports wizards to create forms and reports that relate to one another. You will also use the Grant Access tabs of the Manage Component dialog to implement security at the component level.

You need Edit or higher access privileges to the demo schema to complete this chapter. You also need Edit or higher application privileges to build the components in this chapter. If you do not have these privileges, contact your portal administrator.

The completed portal is available in the installation package that contains the demo schema. For more information, contact your portal administrator.

This chapter contains the following sections:

Feature	Location
Create tabs for a portal and apply security to each.	Section 5.1, "Create three tabs for your intranet portal"

Feature	Location
Use the Create Hierarchy wizard to build a team organization chart, then link to it from an employee lookup form.	Section 5.2, "Create an employee search form that links to an organization chart"
Use the Create From based on Tables or Views wizard to create a job vacancy lookup form and a job posting form, create dynamic and static lists of values, and use the Create Report from SQL Query wizard to create a job vacancies report.	Section 5.3, "Create components to display and post job vacancies"
Create a database sequence, then create two forms: one to request and one to approve a vacation. Also use the Create Calendar wizard to build a calendar that displays requested vacation time, and a report that display remaining vacation time for each employee.	Section 5.4, "Create a vacation request and approval system"
Use the Grant Access tab of the Manage Component dialog to apply security at the component level.	Section 5.5, "Implement security to control display of components"
Use the Customize Page wizard to design the three tabs of the portal and leverage information from an existing content area in the portal.	Section 5.6, "Design the tabs of the HR portal"

5.1 Create three tabs for your intranet portal

In this section, you will add three tabs to the Inside Healthy Living Human Resources portal that pertain only to the Human Resources department. You will create one each for public users, Managers within the company, and HR Representatives. If you did not follow the steps in Chapter 4, "Creating a line of business page for your intranet portal", you must first create a page called "<YourName> Inside Healthy Living," then add three tabs to the page. For more information on creating pages, refer to Section 3.1, "Create a page".

After you create the tabs, you will specify access privileges for each tab. Any user can view the Human Resources tab, only company managers can view the Managers tab, and only HR representatives can view the HR Representatives tab. This way, when a user logs onto Oracle Portal, only one or two of these tabs will display, depending on the type of user. To



implement these access privileges, you need access to the two groups we have provided: HR_MANAGERS and HR_REPRESENTATIVES. If you do not have access to these groups, contact your portal administrator.

Note: All portal administrator tasks are detailed in Appendix A, "Oracle Portal Administrator Tasks".

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Page Tasks** and click **Go**. In the right pane, click **Creating a page**.

1. In Oracle Portal, navigate to the **Pages** tab.
2. Click **My Pages** and find the page called **<YourName> Inside Healthy Living**.

Note: You will only see this page if you have followed the steps in Chapter 4, "Creating a line of business page for your intranet portal". If you have not completed Chapter 4, create a page called "<YourName> Inside Healthy Living", then continue with this section. Unlike the illustrations in this chapter, you will not have a **Main** and **Sales** page in your portal.

3. Next to <YourName> Inside Healthy Living, click **Edit**.
4. On the Portlets tab, next to the **Sales** tab, click .
5. Next to the new tab, click  again.
6. Repeat step 5.
7. Name the first tab you created `Human Resources`.
Note: If you're not sure how to edit the tabs, see Section 4.1, "Create a tabbed page and add a page as a portlet".
8. Click the **Advanced Options** tab.
9. Choose the **Specify Access Settings** radio button.
10. Make sure the **Display Tab to Public Users** check box is selected.
11. Click **Apply**.
12. Click **Close**.
13. Name second tab `Managers`.
14. Click the **Advanced Options** tab.
15. Choose the **Specify Access Settings** radio button.
16. Make sure the **Display Tab to Public Users** check box is clear.
17. In the **Grantee** field, enter `HR_MANAGERS`.

Note: The name you enter here is the group name for the Managers of the Healthy Living Wholesalers company. If you do not see the HR_MANAGERS group, see your portal administrator. Your portal administrator can either import the groups necessary for this chapter, or create them himself. For more information, see Section A.5, "Create three groups for the Human Resources portal" in Appendix A.

18. From the Privileges list, choose **Manage**.

The MANAGE privilege enables all members of the HR_MANAGERS group to modify or delete the page, as well as to grant other groups privileges to the page.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Concepts** and click **Go**. In the right pane, click **Who can access a page**.

19. Click **Add**.

Note: By granting access to a specific group of users, only members of the specified group can view the page.

20. Click **Finish**.

21. Name the third tab HR Representatives.

22. Click the **Advanced Options** tab.

23. Choose the **Specify Access Settings** radio button.

24. Make sure the **Display Tab to Public Users** check box is clear.

25. In the **Grantee** field, enter HR_REPRESENTATIVES.

26. From the Privileges list, choose **Manage**.

27. Click **Add**.

28. Click **Apply**.

29. Click **Close**.

30. Click **Finish**.

You have now added three more tabs to the Inside Healthy Living portal you created in Chapter 4, "Creating a line of business page for your intranet portal". The Human Resources tab displays to all users who are logged into Oracle Portal; the Managers tab only displays to users who have been added to the HR_Managers group; the HR Representatives tab only displays to users who have been added to the HR_Representatives group.

Note: Your portal administrator can create a user for each group. To test whether your pages display properly for each type of user, contact your portal administrator.

The tabs of the portal should now look something like this:



5.2 Create an employee search form that links to an organization chart

The steps in this section show you how to use the Create Hierarchy wizard to build a company organization chart. You will then create a form that enables users to look up a specific employee within the company database, by employee number, employee name, job title, or department. Then, you will create a link from the form to the team organization chart.

5.2.1 Create a team organization chart

In this section, you will create a hierarchy that displays the structure of the Healthy Living Wholesalers departments. The form you create in Section 5.2.2, "Create an employee search form that links to the team organization chart" will link to this hierarchy, so that when users access the form on the portal, they can click the link to display the hierarchy.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Concepts** and click **Go**. In the right pane, under **Component Types**, click **What are hierarchies?**

1. In the Navigator, go to **Casebook30_Application**.

Note: If you do not see this application, contact your portal administrator. If you are not sure how to use the Navigator, try going through the steps in Chapter 3, "Creating an intranet portal and adding simple portlets".

2. On the Applications tab, next to Create New, click **Hierarchy**.
3. Name the hierarchy <YourName>_GROUPLIST and make sure **Casebook30_Application** is chosen from the Application list.

4. Base the hierarchy on the <DEMO_SCHEMA> .HLW_STAFF table.
Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed.
5. On the Table or View Column tab, from the **Primary Key Column** list, choose **HLW_STAFF.EMPLOYEE_ID**.
6. From the **Parent Key Column** list, choose **HLW_STAFF.MANAGER_ID**.
7. From the **Start With Column** list, choose **HLW_STAFF.MANAGER_ID**.
8. In the **Display Column Expression** field, enter **HLW_STAFF .NAME**.
9. In the **Default Start With Value** field, enter **1140**.
10. On the Display Options tab, under **Common Options**, choose **2** from the **Maximum Child Levels** list.
11. Select the **Show Parent Level** check box.
12. In the **Max Children (Child Level 1)** field, enter **20**.
13. From the **Hierarchy Type** list, choose **HTML Table**.
14. In the **Max Children (Child Level 2)** field, enter **20**.
15. In the **Expire After (minutes)** field, enter **0**.
16. Under **Drill Up/Down Options**, next to Drill Up, from the **Type** list, choose **Text**.
17. From the **Value Type** list, choose **Column**.
18. In the **Value** field, enter **nvl (job, ' (No Title) ')**.
Note: Entering this value concatenates the job position from the HLW_STAFF table to the employee name. If no job title exists for the employee, "No Title" displays under the employee's name in the hierarchy.
19. Under **Full Page Options**, find **Font Face**.
20. Next to Font Face, from the **Parent** list, choose **Tahoma**.
21. From the **Child** list, choose **Tahoma**.
22. From the **Grandchild** list, choose **Tahoma**.
23. Next to Font Size, from the **Parent** list, choose **-1**.
24. From the **Child** list, choose **-1**.
25. From the **Grandchild** list, choose **-2**.

26. Next to Background Color, from the **Parent** list, choose **Green, Pale**.
27. From the **Child** list, choose **White**.
28. Clear the following checkboxes:
 - ☐ **Log Activity**
 - ☐ **Show Timing**
 - ☐ **Show Query Criteria**
29. Under **Portlet Options**, find **Font Face**.
30. Next to Font Face, from the **Parent** list, choose **Tahoma**.
31. From the **Child** list, choose **Tahoma**.
32. From the **Grandchild** list, choose **Tahoma**.
33. Next to Font Size, from the **Parent** list, choose **-1**.
34. From the **Child** list, choose **-2**.
35. From the **Grandchild** list, choose **-3**.
36. Next to Font Color, from the **Parent** list, choose **Black**.
37. From the **Child** list, choose **Blue, Midnight**.
38. From the **Grandchild** list, choose **people_color**.
39. Next to Background Color, from the **Parent** list, choose **Green, Pale**.
40. From the **Child** list, choose **White**.
41. On the Hierarchy and Customization Form text tab, in the **Display Name** field, enter
Team Structure.
42. In the **Customization Form** field, enter Customize Hierarchy.
43. In the **Header Text** field, enter:

```
<b><font face="Arial"><font color="#002299"><font size="+2"> The HLW Group
</font></font></font></b>
```
44. In the **Footer Text** field, enter:

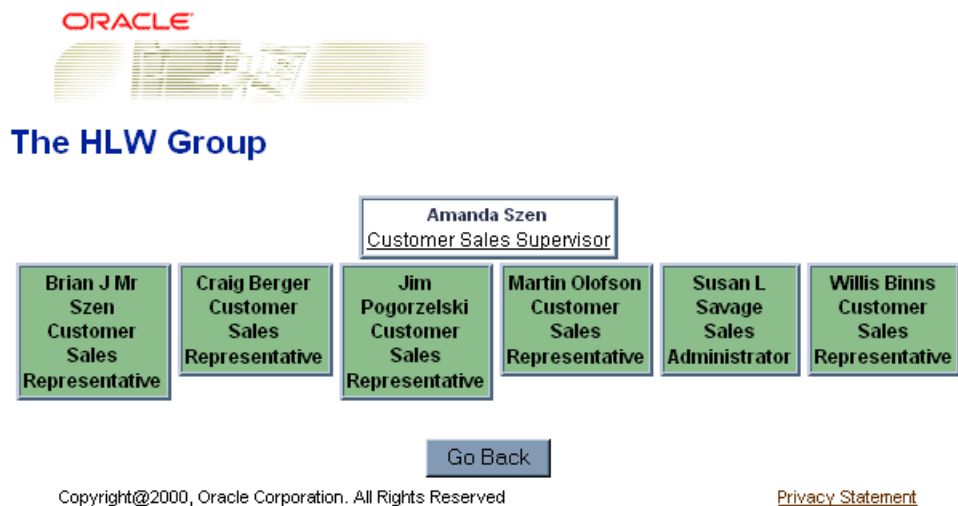
```
<div align=center>
<form>
<font size="+2">
<input type=button value="Go Back" onClick="javascript:history.go(-1);">
```

Note: This footer text adds a "Back" button to the hierarchy so that you can return to the portal once you've clicked on the hierarchy.

45. Finish the wizard and publish the hierarchy to the portal.

Tip: To publish a component to the portal, go to the Manage Component dialog and click the **Access** tab. Select the **Publish to Portal** check box, then click **Apply**.

When you run the hierarchy, a new browser should display a department's organization that looks something like this:



5.2.1.1 Create a company-wide organization chart

If you want to create an organization chart for your entire company, follow these steps. We do not include this hierarchy in the Inside Healthy Living portal.

1. Create a hierarchy in **Casebook30_Application**.
2. Name the hierarchy <YourName>_ALLGROUPLIST and make sure **Casebook30_Application** is chosen from the Application list.
3. Follow steps 4 through 9 in Section 5.2.1, "Create a team organization chart".
4. In the **Default Start With Value** field, enter 30.
5. Follow steps 11 through 46 in Section 5.2.1, "Create a team organization chart".
6. In the **Display Name** field, enter Company Structure.

7. In the **Customization Form** field, enter `Customize Hierarchy`.
8. In the **Header Text** field, enter:

```
<b><font face="Arial"><font color="#002299"><font size=+2> The HLW Group
</font></font></font></b>
```
9. Follow step 50 in Section 5.2.1, "Create a team organization chart", then click **Finish**.

You have created a hierarchy component that displays a company's organization.

5.2.2 Create an employee search form that links to the team organization chart

The steps in this section show you how to create a form to search for specific employees in the Healthy Living Company. You will also create a link that, when clicked, displays the hierarchy you created in Section 5.2.1, "Create a team organization chart".

1. In **Casebook30_Application**, create a form based on a table or view.
Note: If you do not see this application, contact your portal administrator. The steps in Appendix A, "Oracle Portal Administrator Tasks" assist in obtaining the application we have created for you. For more information on creating a form based on a table or view, see Section 4.6.2, "Create a customer entry form".
2. Name the form `<YourName>_EMPSEARCH` and make sure **Casebook30_Application** is chosen from the Application list.
3. Base the form on the `<DEMO_SCHEMA> .HLW_STAFF` table.
Note: `DEMO_SCHEMA` is the name of the schema where the demo schema objects are installed.
4. Choose a **Tabular** Form layout.
5. On the Formatting and Validation Options tab, in the left pane, delete the all buttons *except*:
 - n `QUERY_TOP`
 - n `RESET_TOP`
6. In the left pane, under `TOP_SECTION`, delete all columns in the table, *except*:
 - n `EMPLOYEE_ID`
 - n `NAME`
 - n `JOB`
 - n `DEPARTMENT_NO`

7. Make the following changes to the **Form Level Options**:
 - n **Box Background Color** = Wheat
 - n **Box Background Image** = Dude
 - n **Box Border** = Thick Border
 - n **Order by** NAME, Ascending
8. Change the font, font color, and font size for all fields to:
 - n **Font Face** = Arial
 - n **Font Color** = Black
 - n **Font Size** = 0
9. Change the label for **QUERY_TOP** to Search.
10. Change the label for **EMPLOYEE_ID** to Employee#.
11. In the left pane, click **JOB**.
12. From the **Item Type** list, choose **Combobox**.
13. In the **Label** field, enter Position.
14. From the **List of Values** list, choose <YourName>_JOB_LIST.

Note: This is the LOV you created in Section 4.5.3.1, "Create JOB_LIST dynamic list of values". If you did not create this LOV, you can create it now or use **HLW_JOBLIST**, which we have provided. If you do not see HLW_JOBLIST, contact your portal administrator.
15. Make sure the **Default Value** field is empty, and that **Default Value Type** is **No Selection**.
16. In the left pane, click **DEPARTMENT_NO**.
17. From the **Item Type** list, choose **Combobox**.
18. In the **Label** field, enter Department.
19. From the **List of Values** list, choose <YourName>_DEPARTMENT_LOV.

Note: This is the LOV you created in Section 4.5.3.3, "Create DEPARTMENT_LOV static list of values". If you did not create this LOV, use **HLW_DEPT_LOV**, which we have provided. If you do not see HLW_DEPT_LOV, contact your portal administrator.
20. Make sure the **Default Value** field is empty, and that **Default Value Type** is **No Selection**.

Note: If you do not want to link the employee search form to the team organization chart, you can click **Finish** at this point, then jump to step 32.

21. Next to FORM, click  .

A dialog displays.

22. In the **Name** field, enter `Contact Employee`.

23. Click **OK**.

24. Make the following changes to the Item Level Options for **Contact Employee**.

ⁿ **Item Type** = Label Only

ⁿ **Label** = The HLW Group Organization

ⁿ **Link** = `<DEMO_SCHEMA>.<YourName>_GROUPLIST.show`.

Note: This URL is for the hierarchy you created in Section 5.2.1, "Create a team organization chart". Oracle Portal home is the URL for Oracle Portal. DEMO SCHEMA is name of the schema we provided to you. See your portal administrator for more information.

ⁿ **Font Face** = Arial

ⁿ **Font Color** = Red

ⁿ **Font Size** = +1

25. Finish the form and publish it to the portal.

When you run the employee search form with a link to the hierarchy, it should look something like this:

5.3 Create components to display and post job vacancies

The steps in this section show you how to create a report and two forms. One form enables a public user to look up a particular vacancy in the database to determine whether it has been filled. The second form enables an HR representative to post or close a job vacancy. The report displays all current job vacancies.

5.3.1 Create a form to look up a specific job vacancy

In this section, you will create a form that will be connected to the job vacancy report. After you create this form, you will create the report, then link to this form from the report.

1. In **Casebook30_Application**, create a form based on tables or views.

Note: For more information on creating a form based on a table or view, see Section 5.2.2, "Create an employee search form that links to the team organization chart". If you do not have access to Casebook30_Application, contact your portal administrator.

2. Name the form `<YourName>_VACANCYLOOKUP` and make sure **Casebook30_Application** is chosen from the Application list.
3. Base the form on `<DEMO_SCHEMA>.HLW_VACANCIES`.

Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed.

4. On the Formatting and Validation Options tab, remove the following columns from the form:

- DEPARTMENT_NO

- n REGION_ID
 - n HIRING_MANAGER
 - n HR_CONTACT
 - n VACANCY_STATUS
5. Remove the following buttons from the **BOTTOM_SECTION**:
- n INSERT_BOTTOM
 - n UPDATE_BOTTOM
 - n DELETE_BOTTOM
 - n QUERY_BOTTOM
 - n RESET_BOTTOM
6. Change the following **Form Level Options**:
- n **Box Background Color** = Wheat
 - n **Box Background Image** = Dude
 - n **Box Border** = Thick Border
 - n **Order By** = VACANCY_ID and VACANCY_TITLE
7. Choose VACANCY_ID in the left pane, then change its **Label** to Vacancy#.
8. Choose VACANCY_TITLE in the left pane, then change its label to Position.
9. Change the font and font color for the following fields to **Arial, Black**.
- n VACANCY_ID
 - n VACANCY_TITLE
 - n RESPONSIBILITIES
 - n QUALIFICATIONS
10. On the Form Text tab, in the **Header Text** field, enter:
- ```
 Look Up a Vacancy

```
11. In the **Footer Text** field, enter:
- ```
<div align=center><form><font size=+2><input type=button value="Go Back"
onClick = "javascript:history.go(-1);">
```


Note: This footer text adds a "Back" button to the form so that you can return to the portal after entering information in the form.

12. Finish the form and publish it to the portal.

When you run the job vacancy lookup form, it should look something like this:

Look up a vacancy

Position Details

Vacancy# 1900

Position

Responsibilities

Qualifications

Go Back

5.3.2 Create a form to post or close a job vacancy

The steps in this section show you how to create three LOVs and a form that enables a Human Resources representative to post or close a job vacancy.

5.3.2.1 Create a database sequence for the form

In this section, you will create a database sequence that the job posting form will use when entering records into the database. This sequence will attach an ID number to each record.

1. In the demo schema, create a database sequence for this form.

Note: For more information on creating a database sequence, see Section 3.4.4, "Create a database sequence for the form".

2. Name the database sequence <YourName>_Vacancy_Seq and make sure the demo schema owns the sequence.

3. On the Start and Increment Values tab, in the **Increment By** field, enter 1.
4. In the **Start With** field, enter 10.
5. Finish the wizard and grant the public ALTER access to the sequence.

Tip: After you've finished the wizard, you'll see the new sequence in the Navigator. Next to the sequence name, click **Grant Access**. Add the public as a user with ALTER access privileges.

5.3.2.2 Create three LOVs

5.3.2.2.1 Create the Manager dynamic list of values

This list of values will display all employees with the job titles "Manager" or "Administrator" in the Healthy Living company.

1. In **Casebook30_Application**, create a dynamic list of values.

Note: If you are not familiar with creating a dynamic list of values, see Section 4.5.3.1, "Create JOB_LIST dynamic list of values".

2. Name the list of values <YourName>_MGRLOV.
3. From the **Default format** list, choose **Combo box**.
4. From the **Show Null Value** list, choose **Yes**.
5. In the **SQL Query** field, enter something like this:

```
select NAME, employee_id
from   <DEMO_SCHEMA>.hlw_staff
where  job like '%Manager%'
or     job like '%Administrator%'
and    department_no = :depno
```

Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed.

6. Click **Finish**.

5.3.2.2.2 Create the HR_CONTACTLOV dynamic list of values

This list of values will display all employees in the Human Resources department of the Healthy Living Company. These employees are the HR Representatives.

1. In **Casebook30_Application**, create a dynamic list of values called <YourName>_HRCONTACTLOV.

2. From the **Default format** list, choose **Combo box**.
3. From the **Show Null Value** list, choose **Yes**.
4. In the **SQL Query** field, enter something like this:

```
select name, employee_id from <DEMO_SCHEMA>.hlw_staff where department_no = 70
```

Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed.

5. Click **Finish**.

5.3.2.2.3 Create the Vacancy_Status static list of values

This static list of values will enable a user to select whether the job vacancy is open or closed.

1. In **Casebook30_Application**, create a static list of values called <YourName>_VACANCYSTATUS.
2. From the **Default format** list, choose **Combo box**.
3. From the **Show Null Value** list, choose **No**.
4. In the first row, under **Display Value**, enter Open.
5. Under **Return Value**, enter O.
6. Under **Display Order**, enter 1.
7. In the second row, under **Display Value**, enter Closed.
8. Under **Return Value**, enter C.
9. Under **Display Order**, enter 2.
10. Click **Finish**.

5.3.2.3 Create the form to post or close job vacancies

In this section, you will create a form (which you will later secure so that only HR Representatives can use) that enables a user to post a new job vacancy and update or close an existing one.

1. In **Casebook30_Application**, create a form based on a table or view.

Note: For more information on creating a Form based on a table or view, see Section 5.3.1, "Create a form to look up a specific job vacancy". If you do not have access to Casebook30_Application, contact your portal administrator.

2. Name the form <YourName>_VACANCYFORM and make sure **Casebook30_Application** is chosen from the Application list.

3. Base the form on the <DEMO_SCHEMA>.HLW_VACANCIES table.

Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed.

4. On the Formatting and Validation Options tab, delete the **REGION_ID** field.
5. Change the following Form Level Options:

- **Box Background Color** = Wheat

- **Box Background Image** = Dude

- **Box Border** = Thick Border

6. In the left pane, click **VACANCY_ID** and change its label to Vacancy#.
7. In the **Default Value** field, enter #<DEMO_SCHEMA>.<YourName>_vacancy_seq.nextval.

Note: DEMO_SCHEMA is the name of the schema that owns the database sequence you created in Section 5.3.2.1, "Create a database sequence for the form".

8. From the **Form Level Validation** list, choose **isNumber**.
9. Change the label for **VACANCY_TITLE** to Position.
10. In the left pane, click **DEPARTMENT_NO**.
11. From the **Item Type** list, choose **Combobox**.
12. In the **Label** field, enter Department.

13. From the **List of Values** list, choose <YourName>_DEPARTMENT_LOV.

Note: This is the LOV you created in Section 4.5.3.3, "Create DEPARTMENT_LOV static list of values". If you did not create this LOV, use **HLW_DEPT_LOV**, which we have provided. If you do not see HLW_DEPT_LOV, contact your portal administrator.

14. Make sure the field **Default Value** is empty and **Default Value Type** is **No Selection**.
15. In the left pane, click **HIRING_MANAGER**.
16. From the **Item Type** list, choose **Combobox**.

17. In the **Label** field, enter `Hiring Manager`.
18. From the **List of Values** list, choose `<YourName>_MGRLOV`.
Note: This is the LOV you created in Section 5.3.2.2.1, "Create the Manager dynamic list of values".
19. From the **Set the binding(s) for the bind variable(s) defined in the LOV depno** list, choose `DEPARTMENT_NO`.
Note: For more information on using bind variable LOVs, see Section 4.5.3, "Create four LOVs for the commission search form".
20. Make sure the field **Default Value** is empty and **Default Value Type** is **No Selection**.
21. In the left pane, click `HR_CONTACT`.
22. From the **Item Type** list, choose **Combobox**.
23. In the **Label** field, enter `HR Contact`.
24. From the **List of Values** list, choose `<YourName>_HRCONTACTLOV`.
Note: This is the LOV you created in Section 5.3.2.2.2, "Create the HR_CONTACTLOV dynamic list of values".
25. Make sure the field **Default Value** is empty and **Default Value Type** is **No Selection**.
26. In the left pane, click `VACANCY_STATUS`.
27. From the **Item Type** list, choose **Combobox**.
28. In the **Label** field, enter `Status`.
29. From the **List of Values** list, choose `<YourName>_VACANCYSTATUS`.
Note: This is the LOV you created in Section 5.3.2.2.3, "Create the Vacancy_Status static list of values".
30. Make sure the field **Default Value** is empty and **Default Value Type** is **No Selection**.
31. Change the **Input Width** and **Input Height** to 1.
32. In the left pane, delete all buttons *except*:
 - `NEXT`
 - `PREVIOUS`
33. Change the font face, font color, and font size for all fields to:
 - **Font Face** = Arial

n **Font Color** = Black

n **Font Size** = 0

34. On the Form Text tab, in the **Display Name** field, enter Advertise or Close a Vacancy.

35. In the **Header Text** field, enter:

```
<b><font face="Arial"><font color="#002299"><font size=+0> Advertise a  
Vacancy  
</font></font></font></b>
```

36. In the **Footer Text** field, enter:

```
<div align=center>  
<form>  
<font size=+2>  
<input type=button value="Go Back" onClick="javascript:history.go(-1);">
```

Note: This footer text adds a "Back" button to the form so that you can return to the portal after entering information in the form.

37. Finish the form and publish it to the portal.

When you run the form to post or close job vacancies, it should look something like this:

Advertise a Vacancy

Position Details

Create Query Reset

Vacancy# 1901

Position

Department

Hiring Manager

HR Contact

Responsibilities

Qualifications

Status Open

Go Back

5.3.3 Create a link to connect the vacancy report with the lookup form

1. In **Casebook30_Application**, create a link.
Note: For more information on creating a link, see Section 3.4.5, "Create a link between the form and the calendar".
2. Name the link **<YourName>_VacancyLink** and make sure **Casebook30_Application** is chosen from the Application list.
3. On the Link Target Type and Name tab, select the **Oracle Portal Component** radio button.
4. In the **Target Component or URL** field, enter **Casebook30_Application.<YourName>_VACANCYLOOKUP**.

Tip: Enter the name of the form you created in Section 5.3.1, "Create a form to look up a specific job vacancy".

5. Click **Finish**.

5.3.4 Create a report that displays all current job vacancies

1. In **Casebook30_Application**, create a report from SQL query.

Note: For more information on creating a report from a SQL query, see the online help.

2. Name the report `<YourName>_VACANCIESREPORT` and make sure **Casebook30_Application** is chosen from the Application list.
3. On the SQL Query tab, in the SQL Query box, enter:

```
select vacancy_id VacID,  
       vacancy_title Position,  
       decode (vacancy_status, 'O', 'Open', 'C', 'Closed') status  
from   <demo_schema>.hlw_vacancies  
where  vacancy_status = 'O'
```

Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed.

4. On the Report Layout tab, choose a **Tabular** layout.
5. On the Column Formatting tab, next to **VACID**, enter `Vacancy#`.
6. From the **Align** list, choose **Right**.
7. From the **Link** list, choose `<YourName>_VacancyLink`.
Note: Choose the link you created in Section 5.3.3, "Create a link to connect the vacancy report with the lookup form".
8. Edit the link.
9. In the Set Link Parameters dialog, set `VACANCY_ID = Null`, and choose **VACID** from the Column Name list.
10. Click **Apply**.
11. Click **Close**.
12. Next to **Position**, make sure the Column Heading Text is `Position`.
13. Next to **Status**, make sure the Column Heading Text is `Status`.

14. On the Display Options tab, under **Full Page Options**, change the options to the following:

- n **Type Face** = Tahoma
- n **Font Size** = +0
- n **Border** - Thin Border
- n **Font Color** = Black
- n **Heading Background Color** = Green, Pale
- n **Table Row Color** = White

15. Follow step 15 to make the same changes to **Portlet Options**.

16. Under **Row Order Options**, next to Order by 1., choose **VACID** from the list.

17. On the Report and Customization Form tab, in the **Footer Text** field, enter:

```
<b><font face="Arial"><font color="#002299"><font size=-1> Click on a link  
for position details.</font></font></font></b>
```

18. Finish the report and publish it to the portal.

When you run the job vacancies report, it should look something like this:

Current Open Positions		
Vacancy#	Position	Status
<u>1</u>	Analyst Programmer	Open
<u>1000</u>	Product Manager	Open
<u>1024</u>	Principal Analyst	Open
<u>1100</u>	Analyst	Open
<u>1106</u>	Analyst	Open
<u>1110</u>	Analyst	Open
<u>1142</u>	Chief Database Administrator	Open
<u>1144</u>	Course Developer	Open
<u>1463</u>	Lead Desktop Administrator	Open

[Click on a link for position details.](#)

5.4 Create a vacation request and approval system

The steps in this section show you how to create a form for employees (public users) to request a vacation, a report that displays employees' vacation balances (for managers and HR representatives), and a calendar that displays all employees' vacations with a link to a form where managers can approve the vacation requests.

5.4.1 Create a vacation request form

The steps in this section show you how to create a form where employees can request vacation time. The time requested will then display on a vacation calendar.

5.4.1.1 Create a database sequence for the form

In this section, you will create a database sequence that the job posting form will use when entering records into the database. This sequence will attach an ID number to each record.

1. In the demo schema, create a database sequence for this form.

Note: For more information on creating a database sequence, see Section 3.4.4, "Create a database sequence for the form".

2. Name the database sequence `<YourName>_Vacation_Seq` and make sure the demo schema owns the sequence.
3. On the Start and Increment values tab, in the **Increment By** field, enter 1.
4. In the **Start With** field, enter 10.
5. Finish the wizard and grant the public ALTER access to the sequence.

Tip: After you've finished the wizard, you'll see the new sequence in the Navigator. Next to the sequence name, click **Grant Access**. Add the public as a user with ALTER access privileges.

5.4.1.2 Create EMP_LIST LOV

The steps in this section show you how to create an LOV that will then display as pop-up list in the Vacation Request form you'll create in the next section.

1. In **Casebook30_Application**, create a dynamic list of values called `<YourName>_EMP_LIST`.
2. From the **Default format** list, choose **Pop up**.
3. From the **Show Null Value** list, choose **%**.
4. In the **SQL Query** field, enter something like this:

```
select NAME ,  
       employee_id  
from   <DEMO_SCHEMA>.hlw_staff  
order  by name
```

Note: DEMO_SCHEMA is the name of the schema where the demo schema objects are installed.

5. Click **Finish**.

5.4.1.3 Create the Vacation Request form

The steps in this section show you how to create a form where employees can request vacation time.

You will also use the database sequence you used in Section 5.4.1.1, "Create a database sequence for the form".

1. In **Casebook30_Application**, create a form based on tables or views.
2. Name the form **<YourName>_VACATION_REQUEST** and make sure **Casebook30_Application** is chosen from the Application list.
3. Base the form on the **<DEMO_SCHEMA> .HLW_VACATIONS** table and choose a **Tabular** layout.
4. On the Formatting and Validation Options tab, change the following Form Level Options:
 - **Box Background Color** = Wheat
 - **Box Background Image** = Dude
 - **Box Border** = Thick Border
 - **Order By** = VACA_START_DATE
5. In the left pane, click **VACA_ID** and change its label to **Vacation ID**.
6. In the **Default Value** field, enter **#<DEMO_SCHEMA>.<YourName>_Vacation_Seq.nextval**.

Note: DEMO_SCHEMA is the name of the schema where Oracle Portal is installed. **<YourName>_Vacation_Seq** is the name of the database sequence you created in Section 5.4.1.1, "Create a database sequence for the form".
7. In the **Label** field, enter **Department Number**.
8. In the left pane, click **EMPLOYEE_ID**.

9. From the **Item Type** list, choose **Popup**.

10. In the **Label** field, enter Employee.

11. From the **List of Values** list, choose **<YourName>_EMP_LIST**.

Note: This is the LOV you created in Section 5.4.1.2, "Create EMP_LIST LOV".

12. Change the following item labels:

`VACA_START_DATE` = Start Date

`VACA_END_DATE` = End Date.

`VACA_LENGTH` = Duration.

13. Change the font, font color, and font size for all fields to:

`Font Face` = Arial

`Font Color` = Black

`Font Size` = 0

14. Under **BOTTOM_SECTION**, delete all the buttons *except*:

 PREVIOUS

 NEXT

15. On the Form Text tab, in the **Display Name** field, enter Vacation Request.

16. In the **Header Text** field, enter:

```
<b><font face="Arial"><font color="#002299"><font size=+0> Apply for a  
Vacation</font></font></font></b>
```

17. In the **Footer Text** field, enter:

```
<div align=center>  
<form>  
<font size=+2>  
<input type=button value="Go Back" onClick="javascript:history.go(-1);">
```

Note: This footer text adds a "Back" button to the form so that you can return to the portal after entering information in the form.

18. Finish the form and publish it to the portal.

When you run the vacation request form, it should look something like this:

Vacation Request

Apply for a Vacation

Insert Query Reset

Vacation ID 1700

Employee

Start Date

End Date

Duration

Go Back

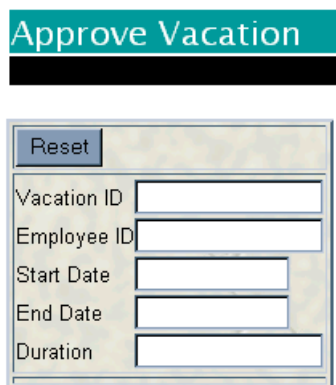
5.4.2 Create a vacation approval form

The steps in this section show you how to create a form that enables a manager to approve an employees vacation request.

1. In **Casebook30_Application**, create a Form based on Tables or Views.
2. Name the form <YourName>_APPROVE_VACATION and make sure **Casebook30_Application** is chosen from the Application list.
3. Base the form on the <DEMO_SCHEMA> .HLW_VACATIONS table.
Note: DEMO_SCHEMA is the name of the schema that owns Oracle Portal.
4. On the Formatting and Validations tab, change the following Form Level Options:
 - **Box Background Color** = Wheat
 - **Box Background Image** = Dude
 - **Box Border** = Thick Border
 - **Order By** = VACA_START_DATE
5. In the left pane, delete all buttons *except*:

- n DELETE_TOP
 - n RESET_TOP
 - n PREVIOUS
 - n NEXT
6. Change the following item labels:
- n **DELETE_TOP** = Reject Request
 - n **VACA_ID** = Vacation ID
 - n **EMPLOYEE_ID** = Employee ID
 - n **VACA_START_DATE** = Start Date
 - n **VACA_END_DATE** = End Date
 - n **VACA_LENGTH** = Duration.
7. Change the font, font color, and font size for all fields to:
- n **Font Face** = Arial
 - n **Font Color** = Black
 - n **Font Size** = 0
8. On the Form Text tab, in the **Display Name** field, enter Approve Vacation.
9. Finish the form and publish it to the portal .

When you run the vacation approval form, it should look something like this:



The screenshot shows a web form titled "Approve Vacation". At the top left is a "Reset" button. Below it are five input fields, each with a label to its left: "Vacation ID", "Employee ID", "Start Date", "End Date", and "Duration". The form has a light blue border and a white background.

5.4.3 Create a calendar that displays requested vacation dates

The steps in this section show you how to create a calendar that displays the requested vacation time. You will use an embedded link to enable a manager can click on a link in the calendar to display the approval form you created in Section 5.4.1.3, "Create the Vacation Request form".

1. In **Casebook30_Application**, create a calendar.

Note: For more information on creating a calendar, see Section 3.4, "Create an event calendar and add it as a portlet".

2. Name the calendar <YourName>_VacationCalendar, and make sure **Casebook30_Application** is chosen from the Application list.
3. On the SQL Query tab, in the **SQL Query** field, enter:

```
select
    v.vaca_start_date the_date,
    E.name           the_name,
    null             the_date_link,
    '<DEMO_SCHEMA>.wwa_app_module.link?p_arg_names=_moduleid&p_arg_
values=<REQUEST_FORM_MODULE_ID>&p_arg_names=_sessionid&p_arg_values=&p_arg_
names=_employee_id_cond&p_arg_values=%3D&p_arg_names=_show_header&p_arg_
values=YES'||v.employee_ID||' '           the_name_link,
    null             the_target
from <DEMO_SCHEMA>.hlw_vacations v
    ,<DEMO_SCHEMA>.hlw_staff e
where e.employee_id = v.employee_id
order by 1
```

Note: Do not copy and paste this SQL code directly into your browser. You must change a few parameters first. REQUEST_FORM_MODULE_ID is the module ID for the Vacation Request form you created in Section 5.4.1, "Create a vacation request form". You can find the module ID by going to the Manage Component dialog for the form, then dragging your mouse over the **Run** link. The module ID displays in the status bar at the bottom of your browser. DEMO_SCHEMA is the name of the schema that owns Oracle Portal.

4. On the Display Options tab, in the **Link Icon** field, enter wgen.gif.
5. On the Calendar and Customization Form tab, in the **Footer Text** field, enter:


```
<b><font face="Arial"><font color="#002299"><font size=-1> My Team's Calendar
</font></font></font></b>
```

6. Finish the calendar and publish it to the portal.

When you run the calendar that displays requested vacation time, it should look something like this:

HLW Vacation Calendar

January 2000

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
						<u>01</u>  Jenny Wong
02	03	04	05	06	07	08
09	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	31					

Note: You may need to use the Vacation Request form that you created in Section 5.4.1, "Create a vacation request form" to submit data to your database so that the calendar displays.

5.4.4 Create a vacation balance report

The steps in this section show you how to create a simple report that displays the remaining vacation balances for all employees. The data in this report changes according to what vacation time has been approved by the manager.

1. In **Casebook30_Application**, create a report from SQL query.

Note: For more information on creating a report from a SQL query, see Section 5.3.4, "Create a report that displays all current job vacancies".

2. Name the report <YourName>_VACATION_BALANCE and make sure **Casebook30_Application** is chosen from the Application list.
3. On the SQL Query tab, in the **SQL Query** field, enter:

```
select e.name      name
      ,e.employee_id empno
      ,e.vacation_days vacdays
      ,v.vaca_length vaclen
from   hlw_staff e
      ,hlw_vacations v
where  e.employee_id = v.employee_id
```

4. On the Report Layout tab, choose **Tabular**.
5. On the Column Formatting tab, change the **Column Heading Text** for the following columns:
 - n **NAME** = Name
 - n **EMPNO** = Employee ID
 - n **VACDAYS** = Accumulated
 - n **VACLEN** = Taken
6. On the Display Options tab, make the following changes to the **Full Page Options** and **Portlet Options**:
 - n **Type Face** = Tahoma
 - n **Font Size** = +0
 - n **Border** = Thin Border
 - n **Font Color** = Black
 - n **Heading Background Color** = Green, Pale
 - n **Table Row Color(s)** = White
7. Make the following changes to the Row Order Options:
 - n **Order By** = NAME

8. On the Customization Form Display Options tab, in the **Display Name** field, enter Staff Vacation Time.

9. In the **Footer Text** field, enter:

```
<b><font face="Arial"><font color="#002299"><font size=-1> Current Vacation  
Balances  
</font></font></font></b>
```

10. Finish the report and publish it to the portal.

When you run the report that displays employees' vacation balances, it should look something like this:

Staff Vacation Time			
Name	Employee ID	Accumulated	Taken
Amanda Szen	1140	20	1
Douglas McCormick	1000	8	No time taken
Gregg W Choplin	480	5	1
Jenny Wong	510	21	1
John S Maulik	440	10	2
Leslie Cleveland	620	22	1

Current Vacation Balances

5.5 Implement security to control display of components

The steps in this section show you how to manage the security of the components which will not be accessible by the public. Only implement security for objects that will be used by a select group of people. The security we implement here is based on two existing groups: HR Managers and HR Representatives.

For more information: Visit the *Oracle Portal Online Help*. From the **Categories** list, choose **Application Tasks** and click **Go**. In the right pane, under **Building Components**, click **Granting component access privileges**.

In this section, we will enable HR Managers to view any public components and the following:

Table 5–1 HR Managers Components

Component	To build this component:
Vacation calendar	See Section 5.4.3, "Create a calendar that displays requested vacation dates"
Vacation Approval form	See Section 5.4.2, "Create a vacation approval form".
Vacation Balances report	See Section 5.4.4, "Create a vacation balance report".

In addition to the public components, we will enable HR Representatives to view:

Table 5–2 HR Representatives Components

Component	To build this component:
Advertise a Job Vacancy form	See Section 5.3.2, "Create a form to post or close a job vacancy".
Vacation Balances report	See Section 5.4.4, "Create a vacation balance report".

Contact your portal administrator if you have any questions about the HR Representatives and HR Managers groups.

5.5.0.1 Apply security to the Vacation Calendar

1. In the Navigator, open **Casebook30_Application**.
2. Next to <YourName>_Vacation_Calendar, click **Grant Access**.
3. Clear the **Inherit Privileges from Application** check box.
4. Under Grant Access, in the **Grantee** field, enter HR_MANAGERS.
5. From the Privileges list, select **Customize**.

Note: Choosing Customize enables the members of the HR_MANAGERS group to use the Customize link displayed on the portlets. Clicking the Customize link in the portlet runs the associated Customization Form, where users can change the parameters of the component.

6. Click **Add**.

The group displays under Change Access.

Note: You can edit which group(s) sees this component by using the Privileges list under Change Access.

7. Click **Apply**.

8. Click **Close**.

You can now add the Vacation Calendar to the Manager tab of the Inside Healthy Living portal. For more information, see Section 5.6, "Design the tabs of the HR portal".

5.5.0.2 Apply security to the Vacation Approval Form

1. In the Navigator, open **Casebook30_Application**.

2. Next to <YourName>_Approve_Vacation, click **Grant Access**.

3. Follow steps 3 through 8 in Section 5.5.0.1, "Apply security to the Vacation Calendar".

You can now add the Vacation Approval form to the Manager tab of the Inside Healthy Living portal. For more information, see Section 5.6, "Design the tabs of the HR portal".

5.5.0.3 Apply security to the Vacation Balance Report

1. In the Navigator, open **Casebook30_Application**.

2. Next to <YourName>_Vacation_Balance, click **Grant Access**.

3. Under Grant Access, in the **Grantee** field, enter HR_MANAGERS.

4. From the Privileges list, select **Customize**.

5. Click **Add**.

The group displays under Change Access.

Note: You can edit which group(s) sees this component by using the Privileges list under Change Access.

6. Under Grant Access, in the **Grantee** field, enter HR_REPRESENTATIVES.

7. From the Privileges list, select **Customize**.

8. Click **Add**.

9. Click **Apply**.

10. Click **Close**.

You can now add the Vacation Balance Report to the Manager tab and HR Representatives tab of the Inside Healthy Living portal. For more information, see Section 5.6, "Design the tabs of the HR portal".

5.5.0.4 Apply security to the Advertise Vacancy form

1. In the Navigator, open **Casebook30_Application**.
2. Next to <YourName>_VacancyForm, click **Grant Access**.
3. Under Grant Access, in the **Grantee** field, enter HR_REPRESENTATIVES.
4. From the Privileges list, select **Customize**.
5. Click **Add**.

The group displays under Change Access.

Note: You can edit which group(s) sees this component by using the Privileges list under Change Access.

6. Click **Apply**.
7. Click **Finish**.

You can now add the Advertise Vacancy form to the HR Representatives tab of the Inside Healthy Living portal. For more information, see Section 5.6, "Design the tabs of the HR portal".

5.6 Design the tabs of the HR portal

The steps in this section show you how to to organize the portlets on your pages so that the appropriate component displays on the relevant page. For example, since only managers can approve vacations, the vacation calendar and the relevant approval form display only on the Managers tab.

You will also leverage the folders in an existing content area called the Inside Healthy Living Repository. For more information on creating this content area, see Chapter 6, "Creating an information repository".

5.6.1 Add portlets to the Human Resources tab

The steps in this section show you how to build the main Human Resources page, which is viewable by the public.

1. In the Navigator, next to the **Inside Healthy Living** page, click **Edit**.
2. Click the **Human Resources** tab.
3. Add a column, then add two rows to the left region, and two rows to the right region.

Note: If you are not sure how to add columns, rows, or regions to a tab, see Chapter 3, "Creating an intranet portal and adding simple portlets".

4. In the top left region, add a portlet.
5. In the Add Portlets dialog, find **Healthy Living Repository**.
6. Add the **Human Resources** folder.
7. Click **OK**.
8. In the bottom left region, add **<YourName>_Vacation_Request**.
Note: This is the form you created in Section 5.4.1, "Create a vacation request form".
9. In the top right region, add **<YourName>_VACANCIESREPORT** as a portlet.
Note: This is the report you created in Section 5.3.4, "Create a report that displays all current job vacancies".
10. In the bottom right region, add **<YourName>_Employee_Search** as a portlet.
Note: This is the report you created in Section 5.2.2, "Create an employee search form that links to the team organization chart".
11. Click **Close**.
12. On the **Pages** tab of the Navigator, click **<YourName> Inside Healthy Living** to view the portal. Click the **Human Resources** tab if it's not already highlighted.

The Human Resources tab should look something like this:

The screenshot displays the 'Human Resources' tab of a portal. The top navigation bar includes 'Main', 'Sales', 'Human Resources' (selected), 'Managers', and 'HR Representatives'. The 'Human Resources' portlet contains a 'Welcome' message, a 'More...' link, and links for 'Benefits Information', 'Recruitment and Hiring', and 'Inside Healthy Living Repository'. The 'Vacation Request' portlet includes a form with fields for 'Vacation ID', 'Employee', 'Start Date', 'End Date', and 'Duration', along with 'Insert', 'Query', and 'Reset' buttons. The 'Current Open Positions' portlet displays a table of open positions. The 'Employee Search' portlet includes a search form with fields for 'Employee#', 'Name', 'Position', and 'Department', along with 'Search' and 'Reset' buttons. A link for 'The HLW Group Organization' is also visible.

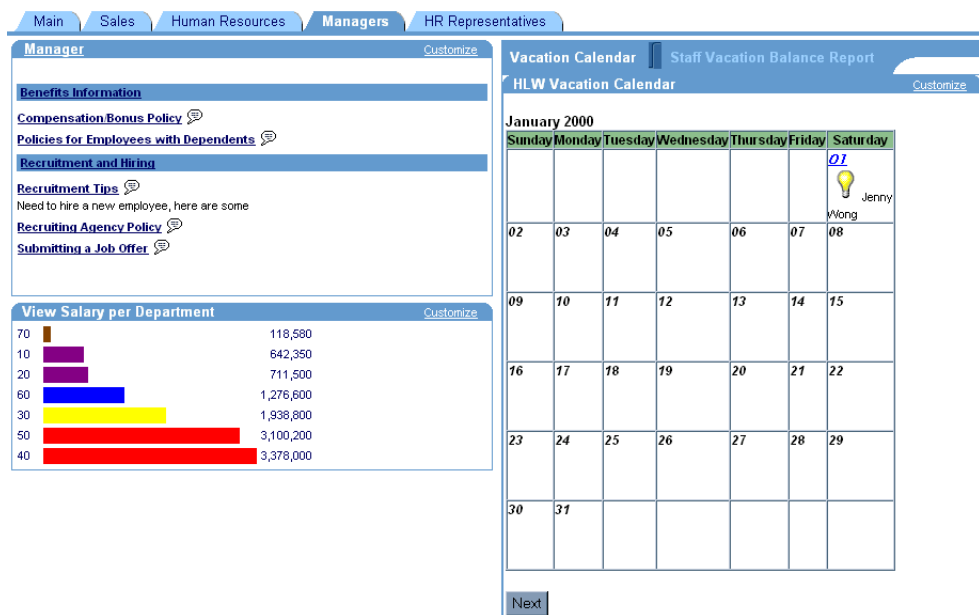
Vacancy#	Position	Status
1	Analyst Programmer	Open
1000	Product Manager	Open
1024	Principal Analyst	Open
1100	Analyst	Open
1106	Analyst	Open
1110	Analyst	Open
1142	Chief Database Administrator	Open
1144	Course Developer	Open
1463	Lead Desktop Administrator	Open

5.6.2 Build the Managers tab

The steps in this section show you how to design the Managers tab of the Inside Healthy Living portal.

1. In the Navigator, next to <YourName> Inside Healthy Living, click **Edit**.
2. On the **Add Portlets** page, click the **Managers** tab.
3. Add a column, then add a row to the left region.
Note: If you are not sure how to add columns, rows, or regions to a tab, see Chapter 3, "Creating an intranet portal and adding simple portlets".
4. In the top left region, add the **Manager** perspective from the Healthy Living Repository (content area).
Tip: This portlet is under the **Inside Healthy living Repository** provider.
5. In the bottom left region, add the **View Salary by Department** chart from the Tutorial30 Application.
Note: If you do not see this chart, contact your portal administrator. Or, you can omit this chart from the tab.
6. In the right region, create a two-tabbed portlet.
7. Name the first tab `Vacation Calendar`.
8. Name the second tab `Staff Vacation Balance Report`.
9. Add `<YourName>_Vacation_Calendar` to the first tab.
Note: This is the component you built in Section 5.4.3, "Create a calendar that displays requested vacation dates".
10. Add `<YourName>_Vacation_Balance` to the second tab.
Note: This is the component you built in Section 5.4.4, "Create a vacation balance report".
11. Click **Close**.
12. On the **Pages** tab of the Navigator, click `<YourName> Inside Healthy Living` to view the portal. Click the **Managers** tab if it's not already highlighted.

The Managers tab should look something like this:



5.6.3 Build the HR Representatives tab

The steps in this section show you how to design the HR Representatives tab of the Inside Healthy Living portal.

1. In the Navigator, next to <YourName> Inside Healthy Living, click **Edit**.
2. On the **Add Portlets** page, click the **HR Representatives** tab.
3. Add a column, then add two rows to the left region.

Note: If you are not sure how to add columns, rows, or regions to a tab, see Chapter 3, "Creating an intranet portal and adding simple portlets".
4. Add one row to the right region.
5. In the top left region, add the **HR Rep** perspective from the Healthy Living Repository (content area).

Tip: This portlet is under the **Inside Healthy living Repository** provider.
6. In the middle left region, add <YourName>_Vacation_Balance.

Note: This is the component you built in Section 5.4.4, "Create a vacation balance report".

7. In the bottom left region, add the **Update Employee Details** form from the Tutorial30 Application.
Note: If you do not see this chart, contact your portal administrator. Or, you can omit this chart from the tab.
8. In the top right region, create a two-tabbed portlet.
9. Name the first tab `Current Job Vacancies`.
10. Name the second tab `Post or Close a Job Vacancy`.
11. Add `<YourName>_VACANCIESREPORT` to the first tab.
Note: This is the component you built in Section 5.3.4, "Create a report that displays all current job vacancies".
12. Add `<YourName>_VacancyForm` to the second tab.
Note: This is the component you built in Section 5.3.2, "Create a form to post or close a job vacancy".
13. In the bottom right region, add `HLW_COMM_REG_CHART` from the Casebook30 Application.
14. Click **Close**.
15. On the **Pages** tab of the Navigator, click `<YourName> Inside Healthy Living` to view the portal. Click the **Managers** tab if it's not already highlighted.

The HR Representatives tab should look something like this:

The screenshot displays the 'HR Representatives' tab in a web application. The sidebar on the left contains links for 'Benefits Information', 'Benefits Providers', 'Recruitment and Hiring', 'Schools and Universities', 'Recruiting Agency Policy', 'Compensation Packages', and 'Immigration Policy'. The main content area is divided into three sections:

- Advertise a Vacancy:** A form with fields for 'Vacancy#' (1920), 'Position', 'Department', 'Hiring Manager', 'HR Contact', 'Responsibilities', 'Qualifications', and 'Status' (Open). Buttons for 'Create', 'Query', 'Reset', and 'Go Back' are present.
- Current Vacation Balances:** A table showing employee vacation balances.
- Update Employee Details:** A form for updating employee information, including 'Employee ID', 'Full Name', 'Address', 'City', 'State', 'Zip Code', 'Phone No', and 'Marital Status'. Buttons for 'Query' and 'Reset' are included.

At the bottom right, there is a 'HLW COMM REG CHART' showing a bar chart of regional data:

Region	Value
Central	1,080,000
Northeast	480,000
Northwest	400,000
Southeast	500,000
Southwest	470,000

5.7 Summary

Congratulations! You have created an intranet portal specific to a department that uses different levels of security to customize the portal. You now know how to:

- Create tabs for a portal and implement security on each tab
- Create a job vacancy portlet where different types of users can either view or post job vacancies
- Create a vacation portlet where employees can request a vacation, then managers can approve them based on a report of vacation balances
- Create an team organization chart using the Create Hierarchy wizard

- Create reports based on SQL queries
- Implement security at the component level and page level
- Leverage content areas in a portal
- Design your portal to demonstrate different functions for each type of employee in your organization

For more information about creating a portal and using security, see the *Oracle Portal Online Help*.

Creating an information repository

This chapter describes how to use Oracle Portal to build a repository of information that contains all types of information, including text, documents, and images.

In this chapter, you will use a content area to create an information repository that provides Healthy Living employees with access to all the information they need in their day-to-day lives within the company.

To complete this chapter, you or your portal administrator needs to create a content area for you to work in. Instructions for this are provided in Section A.6, "Create an Inside Healthy Living Repository". You need to be a content area administrator of this content area.

If you want to complete the final section of this chapter, you also need to be able to publish content area objects as portlets. Ask your portal administrator to grant you these privileges. Instructions are provided in Section A.1.1, "Create a user".

Some of the exercises in the chapter involve using the following files that are provided on the Oracle Technology Network (<http://technet.oracle.com/>):

- Repository content - hlw_repository.zip
- Repository images - hlw_repository_images.zip

The completed content area (Inside Healthy Living Repository) is available in the installation package that contains the demo schema. For more information, contact your portal administrator.

This chapter contains the following sections:

Feature	Location
Upload existing content to a content area	Section 6.1, "Upload content and folder structure as a zip file"
Create categories to classify content	Section 6.2, "Create categories"

Feature	Location
Create perspectives to further classify content	Section 6.3, "Create perspectives"
Create custom types to customize the type of content that can be added to a content area	Section 6.4, "Create custom types"
Organize how content is displayed in a folder	Section 6.5, "Change the layout of a folder"
Control access to content	Section 6.6, "Change folder access privileges"
Make content area objects available for inclusion in a portal	Section 6.7, "Publish objects as portlets"

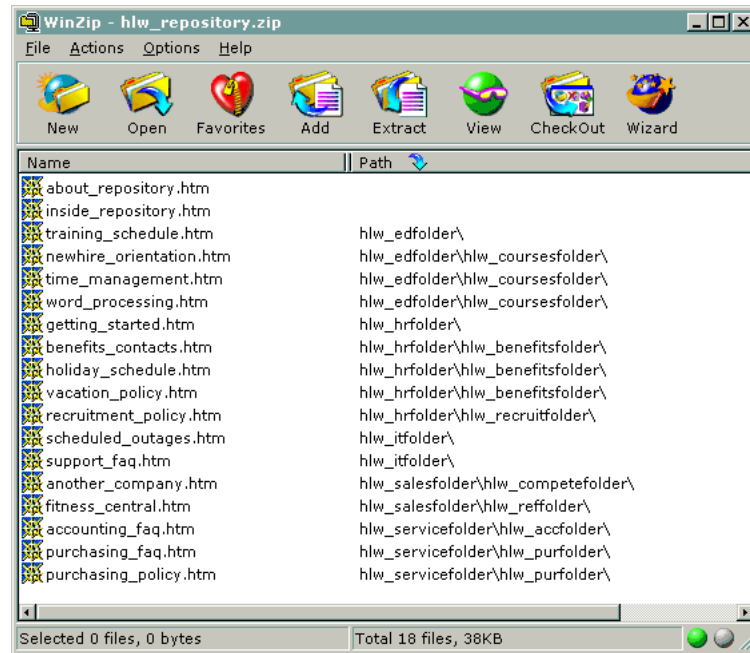
6.1 Upload content and folder structure as a zip file

The steps in this section show you how to upload existing content and structure to a content area. This is extremely useful if you already have content structured within your file system, and don't want to have to create the folder structure in your content area from scratch.

Note: This exercise shows you how to add multiple files to a content area. Any user with the appropriate privileges can also upload individual files to a content area.

For more information: Visit the *Oracle Portal Online Help*. In the navigation bar **Search** field, enter 'adding a file item' and click **Go**.

1. If you haven't already downloaded the Repository content zip file (hlw_repository.zip) from the Oracle Technology Network (<http://technet.oracle.com/>) to your local hard drive, download it now.
2. Open the zip file and take a look at the structure of the files.



You can see that they are already structured within sub-directories.

3. In the Oracle Portal Navigator, click the **Content Areas** tab.
4. Navigate to the root level by clicking **Content Areas** in the Navigator Path, if necessary.
5. Click your content area's name to display the content area.
Note: If you are not sure of the name of your content area, contact your portal administrator.
6. At the top right of the page, click **Edit Folder**.
7. In the folder dashboard, click **Add Item**.
8. In the **Item Type** list, choose **Zip File**.
9. Click **Next**.
10. Next to the **File Name** field, click **Browse**.
11. Locate and select the downloaded Repository content zip file and click **Open**.

12. In the **Display Name** field, enter `Repository content`.

13. Click **Finish**.



14. Now that you have uploaded the zip file, you need to unzip it to add the contents of that zip file to your content area.

Next to the newly added zip file item, click **Unzip**.

15. Make sure the folder in the **Select A Target Folder** field is the root folder of your content area. Any files at the root level of the zip file will be added to this folder and any sub-directories will be created as sub-folders of this folder.

Tip: The name of the root folder is the same as the name of the content area.

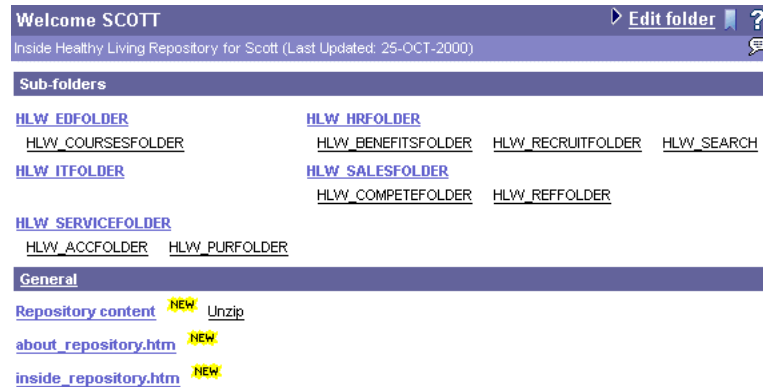
16. Clear the **Run Unzip in Background** check box.

17. Click **Unzip**.



Because you cleared the **Run Unzip in Background** check box, you remain on this page until the file extraction is completed.

18. When the extraction is completed, check the log file to see what has been created and click **Close**.

Note: You may need to refresh the page to display the new contents.



Folders and items have been added to your content area. The folders are created with the same name as the directories in the zip file. The items are created with the same name as the files in the zip file. The folders and items are assigned the same category as the zip file item from which they were extracted.

19. You can change the names displayed for the folders and items by editing them and changing the **Display Name** to something more meaningful. For example:
 - a. At the top right of the page, click **Edit Folder**.
 - b. Click  next to HLW_EDFOLDER.
 - c. In the **Display Name** field, enter `Education`.
 - d. Click **OK**.
20. After unzipping the repository content, you can delete the zip file by clicking  next to the Repository content item.

6.2 Create categories

You can use categories to classify your content and make it easier for users to figure out whether the item is likely to contain the type of information that they are looking for.

Most of the documents added to the repository fall into the following types of information:

- n Lists of contacts
- n Answers to frequently asked questions
- n Documents describing corporate policies

n Schedules of events

The steps in this section show you how to create categories for each of these types of information.

1. In the Navigator, click the **Content Areas** tab.

Tip: If you are currently inside the content area, click **Administration** in the folder dashboard, and click **Navigate Content Area**.

2. Navigate to the root level by clicking **Content Areas** in the Navigator Path, if necessary.
3. Next to your content area, click **Contents** to drill down into the structure of your content area.
4. Next to Categories, click **Create**.
5. In the **Name** field, enter `HLW_CONTACTS`.
6. In the **Display Name** field, enter `Contacts`.
7. Click **Create**.
8. Repeat Steps 5 to 7 to create the remaining categories:


Name	Display Name
HLW_FAQ	FAQ
HLW_POLICY	Policy
HLW_SCHEDULE	Schedule

9. Click **Close**.

6.2.1 Assign a category to an item

Now that you have created your categories, you can associate each of the items in your content area to one of those categories.

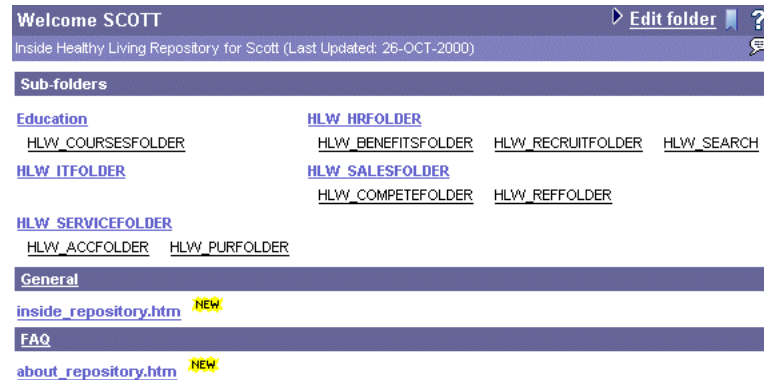
Note: An item can belong to only one category.

1. In the Navigator Path, click **Content Areas** to navigate to the root level.
2. Click your content area's name.
3. At the top right of the page, click **Edit Folder**.
4. Next to the `about_repository.htm` item, click .

5. This document is answering the question 'What is the repository', so the item clearly belongs to the FAQ category.

In the **Category** list, choose **FAQ**.

6. Click **OK**.



The item now displays under a banner that identifies it as belonging to the FAQ category.

If this was your real corporate repository, you would repeat this exercise for each item. However, as this content area is just a sample, you can stop here and leave the rest of the items under the General category.

6.3 Create perspectives

You can use perspectives to further classify content. While categories identify the type of information, perspectives identify the themes of the information, or who would be interested in the information.

The easiest way to identify who would be interested in the information in the repository is to think in terms the user's position within the company. These might include:

- n HR Representative
- n Manager
- n Sales Representative
- n New Employee

You also need a perspective for documents that are useful to all employees. You can also use perspectives to identify the file type of the document, so that users can quickly figure out whether they have the appropriate software or plug-in to view it.

1. In the Navigator, click the **Content Areas** tab.
2. Navigate to the root level by clicking **Content Areas** in the Navigator Path, if necessary.
3. Next to your content area, click **Contents**.
4. Next to Perspectives, click **Create**.
5. In the **Name** field, enter `HLW_ALL`.
6. In the **Display Name** field, enter `All Employees`.
7. Click **Create**.
8. Repeat Steps 5 to 7 to create the remaining perspectives:

Name	Display Name
HLW_HRREP	HR Rep
HLW_HTML	HTML Document
HLW_MANAGER	Manager
HLW_NEW	New Employee
HLW_SALESREP	Sales Rep

9. Click **Close**.

6.3.1 Associate an icon with a perspective

If you want to add a bit more color to your content area, you can associate icons with your perspectives that display alongside items and allow users to see at a glance which perspectives an item belongs to.



1. If you haven't already downloaded the Repository images zip file (`hlw_repository_images.zip`) from the Oracle Technology Network (<http://technet.oracle.com/>) to your local hard drive, download it now.
2. Open the Repository images zip file and extract the file `people.gif`. This image can be used to identify the All Employees perspective.
3. In the Navigator, click **Perspectives**.

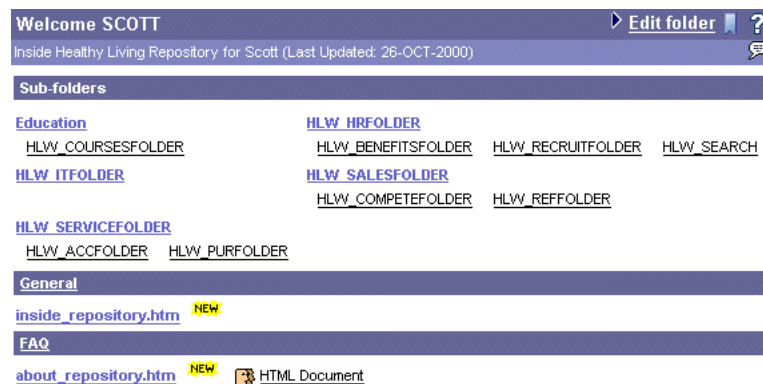
4. Next to All Employees, click **Edit**.
5. Click the **Images** tab.
6. Next to the **Icon Name** field, click **Browse**.
7. Locate and select the extracted `people.gif` file and click **Open**.
8. Click **OK**.

6.3.2 Assign a perspective to an item

Now you have created your perspectives, you can associate each of the items in your content area to one or more of those perspectives.

Note: An item can belong to multiple perspectives.

1. In the Navigator Path, click **Content Areas** to navigate to the root level.
2. Click your content area's name.
3. At the top right of the page, click **Edit Folder**.
4. Next to the `about_repository.htm` item, click .
5. Click the **Secondary** tab.
6. Since all Healthy Living employees will want to read this HTML item, in the **Available Perspectives** list, select **All Employees** and **HTML Document**.
7. Click  to move the selected perspectives to the **Displayed Perspectives** list.
8. Click **OK**.



The perspectives to which the item belongs display next to the item's display name.

Note: Because you associated an icon with the All Employees perspective in the previous exercise, that icon is displayed instead of the text.

6.4 Create custom types

By default, you can add the following types of items to a content area:

- Application Component
- File
- Folder Link
- Image
- Java Application
- PL/SQL
- Text
- URL
- Zip File

The base item types listed above enable you to store a default set of information about each item. To store more specific information about the items in your content area, you can create your own custom item types. First you create attributes to store the information, then create item types to include those attributes.

For example, the Healthy Living Repository includes information about the training courses offered by the Education department. It would be useful to be able to specify information about a specific course, such as its duration and the language in which it is taught, when adding a course description. You can achieve this by creating a Course Description custom item type.


6.4.1 Create attributes

The first thing you need to do when creating a custom item type, is create attributes in which the additional information can be stored. The Course Description item type needs two attributes: one to store the duration of the course, and one to store the language in which the course is taught.

1. In the Navigator, click the **Content Areas** tab.

2. Navigate to the root level by clicking **Content Areas** in the Navigator Path, if necessary.
3. Next to your content area, click **Contents**.
4. Click **Custom Types**.
5. Next to Attributes, click **Create**.
6. In the **Name** field, enter HLW_EDDUR.
7. In the **Display Name** field, enter Duration.
8. The duration of the course can be stored as a simple text string, so in the **Datatype** list, make sure **Text** is chosen.
9. Click **Create**.
10. In the **Name** field, enter HLW_EDLANG.
11. In the **Display Name** field, enter Language.
12. In the **Datatype** list, make sure **Text** is chosen.
13. Click **Create**.
14. A custom attribute is displayed as a field when users add or edit an item of a type that includes the attribute. Instead of expecting users to type in the language of the course, it would be useful to provide them with a list of languages to choose from. You can do this by changing the attribute's display options.

At the top of the page, click **Language**.

15. Select the **List Of Values** radio button.
16. Next to the **List Of Values** field, click  .
17. Click **CASEBOOK30_APPLICATION.HLW_LANGLOV**.

We have provided this List of Values (LOV) for you in the Casebook30_Application. When you want to display an attribute as an LOV, you can create one yourself, or ask an application developer to create one for you.

For more information: Visit the *Oracle Portal Online Help*. In the navigation bar **Search** field, enter 'list of values' and click **Go**.

18. Click **OK**.

6.4.2 Create a custom item type


Now that you have created the attributes in which to store the additional information, you can create the item type that enables users to provide this information.

1. Next to Item Types, click **Create**.
2. In the **Name** field, enter HLW_EDDESC.
3. In the **Display Name** field, enter Course Description.
4. In the **Base Item Type** list, make sure **File** is chosen.

We have provided the full course description as an HTML file. So, you can base the custom item type on the existing File item type to leverage the base attributes required for files.

5. Click **Create**.
6. At the moment, the new item type is exactly the same as the File item type. You now need to edit the Course Description item type to add the extra attributes to it.

At the top of the page, click **Course Description**.

7. Click the **Attributes** tab.
8. In the **Available Attributes** list, select **Duration** and **Language**.
9. Click  to move the selected attributes to the **Selected Attributes** list.
10. Click **Apply**.
11. In the Duration row:

- a. In the **Default Value** field, enter 1 day(s).

Entering a default value means that the Duration field will be prepopulated with this value. Users can accept the default value, or enter a value of their own.

- b. Select the **Primary** check box.

Selecting the **Primary** check box means that this attribute will be displayed on the second page of the wizard when users are adding a Course Description item (or the Primary tab when editing).

12. In the Language row, select the **Primary** check box.


Notice that the Language field is displayed as a list of languages.

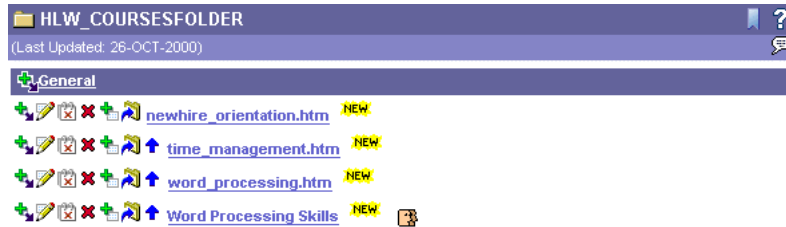
13. Click **OK**.

6.4.3 Add a course description

Let's test out your new item type by adding a course description to the content area.

1. Open the Repository content zip file and extract the file `word_processing.htm`.
2. In the Navigator Path, click **Content Areas** to navigate to the root level.
3. Click your content area's name.
4. Click **HLW_COURSESFOLDER** to drill down into the Course Descriptions folder.
5. At the top right of the page, click **Edit Folder**.
6. In the folder dashboard, click **Add Item**.
7. In the **Item Type** list, choose **Course Description**.
8. Click **Next**.
9. Next to the **File Name** field, click **Browse**.
10. Locate and select the extracted `word_processing.htm` file and click **Open**.
11. In the **Display Name** field, enter `Word Processing Skills`.
12. Notice that the **Duration** field is already populated with the value `1 day(s)`, as you specified when you created the item type.

The actual duration of this course is 2 days, so in the **Duration** field, change 1 to 2.
13. Also notice that there is a **Language** list from which you can choose the language in which the course is conducted. This course is in English, so you can leave this value as it is.
14. Click **Next**.
15. In the **Available Perspectives** list, select **All Employees**.
16. Click  to move the selected perspective to the **Displayed Perspectives** list.
17. Click **Finish**.



6.4.4 Create a custom item type with a procedure call

In addition to using custom item types to store information about items, you can include calls to HTTP and PL/SQL procedures. For example, the HR department wants to include definitions for some benefits terms. It would be useful if, as well as including the definition provided by HR, the definition could also provide the facility to call an external search engine (such as the one for the financial services provider Fidelity) for more detailed information. You can do this by creating a custom item type and adding a call to the search engine.

1. The first thing to do, is to find out how the call to the search engine is constructed, so that you can call it correctly from your item type. You can do this by performing a search and taking a look at the URL generated.

The Fidelity search engine call is constructed as follows:

n **Search engine procedure call:** <http://www.northernlight.com/fidelity/nlquery.fcgi>

n **Fixed parameters:** orl=2:1&dx=8001&db=26&cs=0

n **Search term:** qr=<search term>

2. Now, you need to create an attribute to store the fixed parameters, so that they can be passed to the search engine:
 - a. In the Navigator, click the **Content Areas** tab.
 - b. Navigate to the root level by clicking **Content Areas** in the Navigator Path, if necessary.
 - c. Next to your content area, click **Contents**.
 - d. Click **Custom Types**.
 - e. Next to Attributes, click **Create**.
 - f. In the **Name** field, enter HLW_HRFIXED.

- Note:** You can also upload an image to use as an icon. If you specify an icon, the icon is used instead of the link text.
- You need to pass two things to this procedure: the fixed parameters and the search term.
- In the first **Attribute** list, choose **Fidelity Search Fixed Parameters**.

19. In the corresponding **Pass As** field, enter `orl=2:1&dx=8001&cb=26&cs`.

20. In the second **Attribute** list, choose **Keywords**.

21. In the corresponding **Pass As** field, enter `qr`.

This means that whatever text is entered in the Basic Search Keywords field for an item of this type will be pass as the search term to the Fidelity search engine.

22. Click **OK**.

6.4.5 Add a benefits definition

Let's add a benefits definition to see how the call to the Fidelity search engine works.

1. In the Navigator Path, click **Content Areas** to navigate to the root level.

2. Click your content area's name.

3. Click **HLW_BENEFITSFOLDER** to drill down into the Benefits Information folder.

4. At the top right of the page, click **Edit Folder**.

5. In the folder dashboard, click **Add Item**.

6. In the **Item Type** list, choose **Benefits Definition**.

7. Click **Next**.

8. In the **Text** field, enter `A 401(k) is a type of retirement plan that allows employees to save and invest for their own retirement.`

9. In the **Display Name** field, enter `401(k): Definition`.

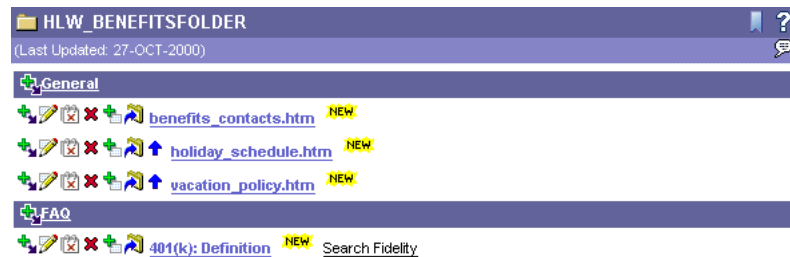
10. In the **Category** list, choose **FAQ**.

11. Click **Next**.

12. Remember, the item type uses the value specified in the Basic Search Keywords field as the term to pass to the Fidelity search engine.

In the **Basic Search Keywords** field, enter `401(k)`.

13. Click **Finish**.



You can see a link next to the item. This is what users will click to launch the Fidelity search engine.

14. Next to the new 401(k) Definition item, click **Search Fidelity** to make the call to the Fidelity search engine that you added to the item type.

A page is returned that lists the contents of the Fidelity Web site that include the term 401(k).

6.5 Change the layout of a folder

So, now you've sorted out the structure of the Healthy Living Repository, created the classification model, and customized the item types to match the content that will be added to the content area. Now it's time to work on the layout of the content area, and make sure that information is easy to locate on the page, and interesting to look at.

A folder is divided into five different regions. You can change how these regions are displayed in the folder. By default the regions are displayed as five rows. Let's change this to something more interesting.

1. In the Navigator, click the **Content Areas** tab.
2. Navigate to the root level by clicking **Content Areas** in the Navigator Path, if necessary.
3. Next to your content area, click **Contents**.
4. The layout of a folder is determined by the style associated with it, so click **Styles**.
5. Next to the style for your content area folders, click **Edit**.

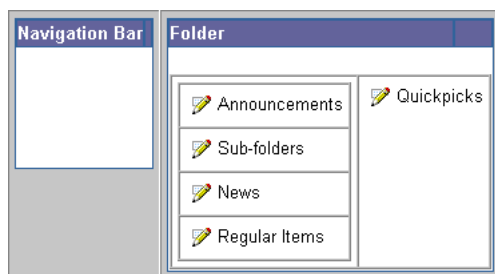
Note: Make sure you edit the style for the content area folders (which has the same name as your content area), and not the style for the content area navigation bars.

The first tab displayed is the one in which you can change the colors and fonts used in the folder. This exercise concentrates on changing the layout of the folder. For information about how to change the colors and fonts to match the Healthy Living look and feel, refer to the *Oracle Portal Tutorial*, which includes step by step instructions on how to do this.

6. Click the **Folder Layout** tab.
7. In the Folder Layout radio group, click the radio button to the left of the following layout:




8. Click **Apply**.



6.5.1 Change region properties


In the previous exercise, you changed how the folder regions are laid out in the folder. You can also define how items are arranged within each region, which item attributes are displayed, and various other display options.

1. The right hand column would be a good region in which to provide frequently requested information, because of its visibility.
Click  to the left of Quickpicks.
2. In the **Display Name** field, enter **FAQ** and **Hints**.
3. At the moment, this region is set so that the banner is not displayed. Let's display the banner in this region, so that users know what type of information it contains.


Select the **Display Region Banner** check box.

4. It would also be useful to display item descriptions in this region, to provide the user with more information about the item.

In the **Available Attributes** list, select **Description**.

5. Click  to move the Description attribute to the **Displayed Attributes** list.
6. But, let's display the description first, so that the description is the first thing the user will read about the item.


In the **Displayed Attributes** list, select **Description**.

7. Click  to move the Description attribute to the top of the list.
8. In the **Item Alignment** list, choose **Left**.
9. In the **Number Of Columns** field, enter 1.
10. Click **Apply**.
11. Click **Close**.
12. Let's use the top region to include a welcome message for the folder.

Click  to the left of Announcements.


13. In the **Display Name** field, enter `Welcome`.
14. Select the **Display Region Banner** check box.
15. Click **Apply**.
16. Click **Close**.




17. Also, let's change the text in the Sub-folders banner to something that will be more meaningful to the users of the Repository, who are not as familiar with the way a content area is broken down into folders and sub-folders.

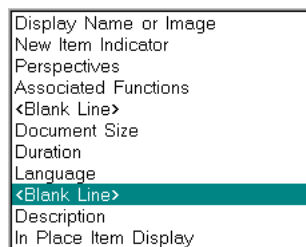
Click  to the left of Sub-folders.

18. In the **Display Name** field, enter `More . . .`.
19. Click **Apply**.
20. Click **Close**.
21. Finally, let's change which attributes to display next to regular items in the folder. For example, it would be useful for users to be able to see the size of files in the content area so that they know how long they are likely to take to download. Also, the Course

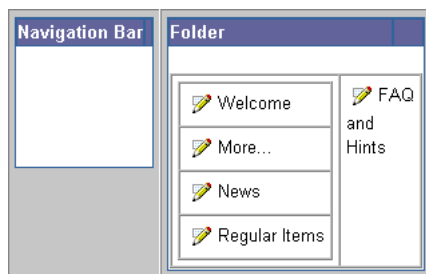
Descriptions custom item type you created earlier includes a couple of attributes that would be useful to display next to course description items.

Click  to the left of Regular Items.

22. In the **Available Attributes** list, select **<Blank Line>**, **Document Size**, **Duration**, and **Language**.
23. Click  to move the selected attribute to the **Displayed Attributes** list.
24. Use  and  to order the attributes in the **Displayed Attributes** list to look like the following.



25. Click **Apply**.
26. Click **Close**.



27. You can also change the way attributes display in the folder.
Click the **Properties** tab.
28. In the **Style Element** list, choose **Item Attribute**.
29. In the **Font Style** list, choose **Italic**.
30. Click **Apply**.

31. Click **OK**.

32. In the Navigator Path, click **Content Areas**.

33. Click your content area's name.

Notice that the banner for the sub-folders now displays the text *More...* instead of *Sub-folders*. Also, the document size attribute for the *about_repository.htm* item is displayed beneath it, in italics.

34. Click **HLW_COURSESFOLDER** to drill down to the Course Descriptions folder.

Notice that the Duration and Language attributes are displayed beneath the Word Processing Skill item you added earlier.

6.5.2 Add an item to a region

Now that you have changed the layout of the folders in the content area, you can add new items directly to the appropriate regions.

You created a Welcome region, so let's add a welcome message for the root folder of the Repository.

1. In the folder path, click your content area's name to return to the root folder.

2. At the top right of the page, click **Edit Folder**.

3. In the folder dashboard, click **Add Item**.

4. In the **Item Type** list, choose **Text**.

5. Click **Next**.

6. In the Folder Region section, select the **Welcome** radio button.

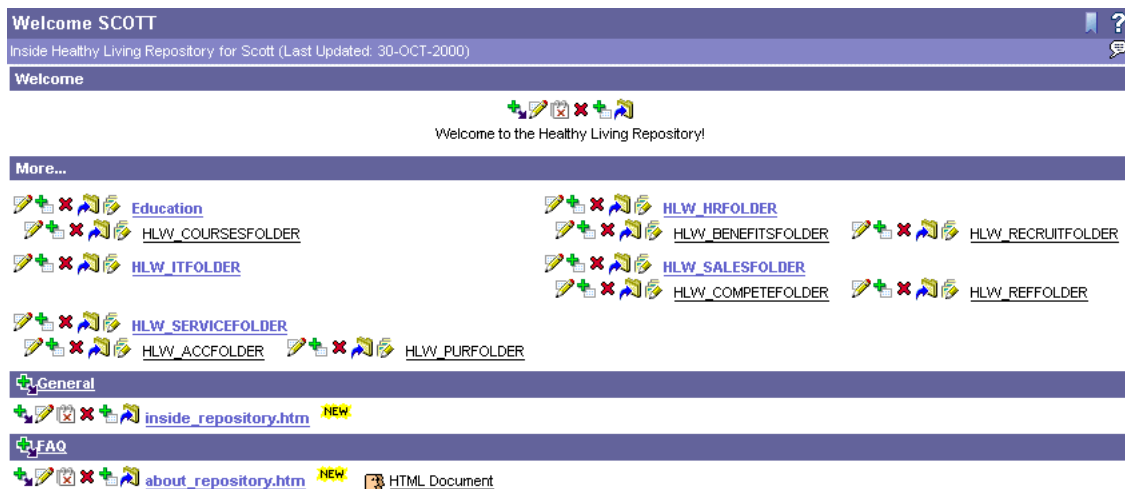
7. In the **Text** field, enter `Welcome to the Healthy Living Repository!`

8. In the **Display Name** field, enter `Welcome!`

9. Click **Next**.

10. Select the **Item Displays Directly In Folder Area** radio button.


11. Click **Finish**.



The Welcome message is displayed at the top of the folder, in the Welcome region.

6.5.3 Move an item to a different region

The Healthy Living Repository already contains several documents. You will need to move these documents into the appropriate regions. For example, you created a region in which to display FAQ documents and there is a FAQ document in this folder: `about_repository.htm`. So, let's move this document into the correct region.

1. Next to the `about_repository.htm` item, click .
2. In the Folder Region section, select the **FAQ and Hints** radio button.
3. In the **Display Name** field, enter `About the Inside Healthy Living Repository`.
4. In the **Description** field, enter `Want to know what the Repository is all about? Take a look at.`
5. Click **OK**.
6. At the top right of the page, click **View Folder**.



The item is now displayed in the FAQ and Hints region. Also, notice that the description of the item is displayed before the item name, just like you specified in the list of displayed attributes when you edited the region properties.

Note: The document size no longer displays next to this item because you did not select to display that attribute in the FAQ and Hints region.

6.6 Change folder access privileges

The next thing you need to consider is who should be able to view the information in the Repository. With content areas you can restrict access to folders, and even to specific items, to particular users or user groups. This is extremely useful for making corporate information available to as wide an audience as possible, while keeping it secure.

1. In the Navigator, click the **Content Areas** tab.
2. Navigate to the root level by clicking **Content Areas** in the Navigator Path, if necessary.
3. Next to your content area, click **Edit Root Folder**.
4. Click the **Access** tab.
5. This folder is the entrance to the Repository and contains general information. It should therefore be available to all users.

Make sure the **Make Public** check box is selected.

6. Most of the folders within the content area should be available to all users. However, when folders are initially created the **Make Public** check box is cleared; only the root

folder is made available to all users. To quickly make all folders available to all users, you can copy the access privileges applied to the root folder to all folders beneath it.

Make sure the **Add Privileges To All Sub-folders** radio button is selected.

7. Click **Cascade Privileges**.
8. Click **OK**.
9. Next to your content area, click **Contents**.
10. Click **Folders**.
11. Next to HLW_HRFOLDER, click **Edit**.
12. Click the **Access** tab.

Notice that the **Make Public** check box is already selected because it uses the same access privileges as defined for the root folder.

13. But what about adding new information? You don't want just anyone to be able to add items to this folder; only specific members of the HR department should be able to do this.

In the **Name** field, enter HR_REPRESENTATIVES.

14. Click **Add To Access List**.

Note: If this group does not exist, contact your portal administrator, or follow the instructions in Section A.5.2, "Create the HR Managers and HR Representatives groups" to create the group.

15. In the HR_REPRESENTATIVES row, select the **Manage Items** check box.

This enables members of the HR_REPRESENTATIVES group to add, edit, and delete any item in this folder.

16. Click **OK**.
17. Next to HLW_HRFOLDER, click **Sub-Folders**.
18. The HLW_RECRUITFOLDER contains information that is applicable only to members of the HR department. It is not information that should be available to all users.

Next to HLW_RECRUITFOLDER, click **Edit**.

19. Click the **Access** tab.
20. Clear the **Make Public** check box.

Now only users with specific View privileges will be able to view the contents of this folder. All other users will not be able to access the folder at all.

21. In the **Name** field, enter HR_REPRESENTATIVES.
22. Click **Add To Access List**.
23. In the HR_REPRESENTATIVES row, make sure only the **View** check box is selected.
24. Click **OK**.

6.7 Publish objects as portlets

You can make certain content area objects available so they can be used in a portal. Because content areas are part of Oracle Portal, publishing objects as portlets is a very simple process. Let's publish the HR Rep perspective as a portlet, so that it can be added to a Healthy Living portal, providing HR Representatives with instant access to those documents of specific relevance to their job.

1. In the Navigator, click the **Content Areas** tab.
2. Navigate to the root level by clicking **Content Areas** in the Navigator Path, if necessary.
3. Next to your content area, click **Contents**.
4. Click **Perspectives**.
5. Next to HR Rep, click **Edit**.
6. Select the **Publish As Portlet** check box.

Note: If you cannot see this check box, you do not have privileges to publish portlets. Ask your portal administrator to grant you these privileges, using the instructions in Section A.1.1, "Create a user".

7. Click **OK**.

You can also publish folders, categories, and navigation bars as portlets in a similar way.

If you want to learn how to add these objects to a portal, take a look at Chapter 5, "Creating a Human Resources portal with security".

6.8 Summary

Congratulations! You have successfully used a content area to create an information repository. You now know how to:

- Upload content to a content area using a zip file

- Create categories and assign categories to items
- Create perspectives, associate icons with perspectives, and assign perspectives to items
- Create custom types
 - Create attributes
 - Create custom item types
 - Add procedure calls to item types
- Change the layout and region properties of folders
- Change folder access privileges
- Publish content area objects as portlets


For more information about content areas, see the *Oracle Portal Online Help*.

Oracle Portal Administrator Tasks

A.1 Creating a user who can build these examples

In this section, you will learn how to create a user in Oracle Portal that can create the portals described in this manual. Note that you must have Develop or higher access privileges to create a user. See your database administrator to obtain this access.

A.1.1 Create a user

1. On the Oracle Portal Home page, click **Administer**.
Note: You can access this page by clicking  in the banner.
2. In the User portlet, click **Create New Users**.
3. In the **User Name** field, enter [new_user_name].
Note: You cannot use spaces or dashes in this field.
4. In the **Password** field, enter a temporary password.
5. In the **Confirm Password** field, enter the same temporary password you entered in Step 4.
6. In the **E-mail Address** field, enter the new user's email address.
7. In the **Account Activation and Termination** section, enter the dates you want this user to be active.
8. In the **Login Server Privileges** section, make sure **End User** is chosen. The portals in this manual do not require the user to access the Login Server.
9. Click **Create**.
10. At the top of the screen, you should see a link for your new user. Click this link.

11. Under Group Membership, select the **PORTLET_PUBLISHERS** check box.
12. Add any other information for the user, such as Job Title or a photograph.
13. Click **Apply**.
14. Click **OK**.

You have now created a user (or users) for Oracle Portal.

A.1.2 Grant the user access to the casebook application

1. In the Navigator, click the **Applications** tab.
2. Navigate to the Casebook30_Application.

Note: The Casebook30_Application is the one into which the sample components were imported (see the Read Me file associated with the installation scripts for more information).
3. Next to Casebook30_Application, click **Grant Access**.
4. In the **Grantee** field, type the name of the user to whom you want to grant access.
5. From the list, choose **Manage**.
6. Click **Add**.
7. Follow steps 4 through 6 for all the other users who will be creating the components in the casebook.
8. Click **Apply**.
9. Click **OK**.

You have now given a user (or users) permission to create components in the Casebook30_Application. The instructions in this book tell the user to preface the names of components they create with their own name. If they name a component with a name that already exists, the existing component may be altered or deleted. You can always re-import the components using the installation scripts.


A.1.3 Enable the user to publish objects as portlets

1. On your Oracle Portal home page, click the **Administer** tab.
2. In the Users portlet:
 - a. In the **Name** field, enter the user name of the user.
 - b. Click **Edit**.

3. Select the **PORTLET_PUBLISHERS** check box.
4. Click **OK**.

A.2 Creating a group

You can also create a group of users to make it easier for you to grant privileges to multiple users simultaneously. Note that you must have Administrator or higher privileges to create a group.

1. On the Oracle Portal Home page, click **Administer**.
Note: You can access this page by clicking  in the banner.
2. In the Groups portlet, click **Create New Groups**.
3. In the **Name** field, enter a group name, for example `Casebook_Group`.
4. Make sure you apply this group to **All of Oracle Portal**.
5. In the Description field, enter: `This group of users is for creating the portals in the Building Advanced Portals manual.`
6. Click **Next**.
7. In the **Name** field, enter the user name of the user you want to add to the group.
8. Click **Add to Members List**. The user name now displays in the Group Member List, along with the user type.
Note: Your user name should already display in the Group Member List since you are the owner of the list.
9. Follow steps 7 and 8 to add other users to the group.
10. Click **Finish**.


You have now created a group of users for Oracle Portal. Note that you can now follow the steps in Appendix A.1.2, "Grant the user access to the casebook application" to enable the group to create components in this application. To do so, type the group's name instead of the user's name in Step 4.

A.3 Installing the demo portals, schema, and content areas

You can access the installation scripts for the demo portals, schema, and content areas at <http://technet.oracle.com>. To install these samples, follow the instructions in the Read Me file located in the same directory on <http://technet.oracle.com> as the installation scripts.

A.4 Grant access privileges to the schema

In order to build a form that updates tables, users may need access privileges to the <DEMO_SCHEMA>. If a user does not have the privileges to grant access to this table, a database administrator (who does have access to the schema) should follow the steps in this section.

1. In the banner at the top of the Oracle Portal home page, click .
2. Click **Database Objects**.
3. Find the <DEMO_SCHEMA> schema.
4. Next to the <DEMO_SCHEMA>, click **Grant Access**.
5. In the **Grantee** field, enter the user name.
6. From the privileges list, select **Manage**.
7. Click **Grant**. The user ID and privileges should now display under Revoke Privileges.
8. Click **Close**.

You have now enabled users to manipulate the tables in the demo schema. Note that the users will be able to modify any data in this schema. If you do not want users to have manage access to all tables in the schema, you can navigate to the individual tables in the schema, then grant individual access to each table.

A.5 Create three groups for the Human Resources portal

The steps in this section help you create three users and three user groups for the purposes of Chapter 5, "Creating a Human Resources portal with security".

A.5.1 Create three users for the groups

1. Create a user with the following characteristics:

Note: If you're not sure how to create a user, follow steps 1 through 9 in Section A.1, "Creating a user who can build these examples".

- n User Name = Joe_Public
- n Password = 12345
- n Confirm Password = 12345

2. Create a second user with the following characteristics:

- n User Name = Joe_Manager
- n Password = 12345
- n Confirm Password = 12345

3. Create a third user with the following characteristics:

- n User Name = Joe_HRRep
- n Password = 12345
- n Confirm Password = 12345

A.5.2 Create the HR Managers and HR Representatives groups

The steps in this section help you create two groups. The Managers group can view the Managers page of the Inside Healthy Living Human Resources portal, and the HR_Representatives group can view the HR Representatives page of the portal.

Note: You do not need to create a specific group to view the Public page of the portal, since any user who logs into Oracle Portal should, by default, be able to see this page.

- 1.** On the Oracle Portal Home page, click the **Administer** tab.
- 2.** In the Groups portlet, click **Create New Groups**.
- 3.** In the **Name** field, enter HR_MANAGERS.
- 4.** Make sure the **All Objects Across Product** radio button is selected.
- 5.** In the **Description** field, enter This group contains the company managers for the Human Resources portal.
- 6.** Under Default Style, choose **Healthy Living Page Style** from the **Display** list.
- 7.** Click **Next**.
- 8.** In the **Name** field, enter Joe_Manager.

Note: This is the user you created in Appendix A.5.1, "Create three users for the groups".

9. Click Add to Members List.

The name you entered displays in the Group Members List.

10. Repeat steps 8 and 9 to add more users to this group.

11. Click Apply.

12. Click OK.

13. Repeat steps 2 through 6 to create a group with the following details:

- **Name** = HR_REPRESENTATIVES
- **Description** = This group contains the Human Resources representatives for the Human Resources portal.

14. Click Next.

15. Follow steps 8 through 12 to add users to this group.

Note: Add the user Joe_HRRep to the group.

You have now create the groups necessary to complete the steps in Chapter 5, "Creating a Human Resources portal with security".

A.6 Create an Inside Healthy Living Repository


1. In the Navigator, click the **Content Areas** tab.
2. Navigate to the root level by clicking **Content Areas** in the **Path**, if necessary.
3. Click **Content Area**.
4. In the Name field, enter a unique name for the content area.

Tip: We recommend including the user name of the user for whom you are creating the content area, to make the name unique. For example, SCOTT_REPOSITORY.

5. In the Display Name field, enter a descriptive name for the content area.

Tip: We recommend including the user name of the user for whom you are creating this content area, to make it easier for the user to find. For example, Inside Healthy Living Repository for Scott.

6. Click **Create**.
7. At the top of the page, click the name of the newly created content area.

8. Click the **Items** tab.
9. Clear the **Retain Deleted Items** and **Display Deleted Items** check boxes.
10. Click the **Access** tab.
11. In the **Grantee** field, enter the user name of the user for whom you created the content area.
Tip: If you are not sure of the user's user name, click  and select the user name from the pop-up window.
12. In the list of privileges, make sure **Administer** is chosen.
13. Click **Add**.
14. Click **OK**.

Glossary

advanced search

Enables content area users to:

- Find content that contains *any* or *all* terms in the search string.
- Search another content area outside the current one, or search across all content areas.
- Restrict the search to a particular folder, category, perspective, item type, or attribute.

If *interMedia* is installed and enabled, advanced search can also be used to perform near, soundex, and fuzzy searches.

API

Application Program Interface. A software program that functions between the operating system and application or between a host application program and client applications.

API, overloaded

Comprised of more than one function or procedure that share same name, but contain differing numbers and/or types of parameters.

application

In Oracle Portal, a group of components connected to one another by links, designed to fulfill a particular business need. For example, a form can be linked a chart that, in turn, allows an end user to drill down to detailed reports about items displayed in the chart.

application component item

One of the item types you can add to a folder. An Oracle Portal application component item may be a chart, form, report, and so on, created by the Oracle Portal Component Build Wizard and stored in the host database. When the user clicks the item, the application component executes and sends the result to the user's browser.

activity log

Database table that provides a record of end user requests for Oracle Portal components. The log includes information about the time of the request, the user who made the request, and the machine and browser type that originated the request.

attribute

Stores information about an item or folder: for example, Create Date, Expire Date, and Author. Content area administrators can create their own attributes to extend the functionality of item types and folder types.

authorized user

User who logs on to Oracle Portal. By default, authorized users can create certain Oracle Portal objects such as folders and pages. Contrast to *public users*, who cannot log on.

banner

A colored, horizontal bar with a title and optional graphic elements displayed on an Oracle Portal page or folder. A banner breaks up the visual flow of a page and helps group related items that appear beneath it.

A folder can contain up to four different types of banners: Main, Sub, Region, and Group By.

base folder type

See *folder type*.

base item type

See *item type*.

basic search

Enables content area users to search the current content area's items, folders, categories, and perspectives for the search string.

batch job

Running a component in the background using the Oracle Portal batch job facility. An end user can run a component in batch mode by selecting options on the customization form for the component. Batch processing is useful if the component is based on a large amount of data, or if the component displays many rows of data.

bind variable

Variable in a SQL statement that must be replaced with a valid value or address of a value in order for the statement to execute successfully. Oracle Portal component developers typically use bind variables to display a parameter entry field in a component's customization form. The entry field allows end users to choose the data that the component will display.

bookmark

See *favorite*.

bulk load

See *Zip file item*.

caching

Temporary storage of a page, folder, or component to reduce performance costs associated with retrieving the object each time it is requested. Users can choose to cache the structure of a page or folder, the structure plus the content, or nothing at all.

calendar

Oracle Portal component that displays the results of a SQL query in calendar format. At least one of the table columns in the query must have the DATE datatype.

call interface

Displays the arguments that were selected when a component was originally created or last edited.

category

A grouping of items in a content area. A category helps users answer the question, "What is this item?" For example, in a travel content area, you might have categories of maps, snapshots, or hotel reviews. Each item added to a content area must be assigned to a category.

chart

A component that displays the results of a SQL query as a bar chart. Charts are based on at least two table or view columns: one that identifies the bars on the chart and another that calculates the size of the bars on the chart.

cluster

Database object used to store tables that are related to one another and that are often joined together in the same area on disk.

color palette

Used to assign colors to elements in a folder style, including background, links, banners, and other items displayed on a folder page. The hex value for each color is displayed in the palette.

component

A PL/SQL stored procedure created by an Oracle Portal component build wizard; for example, a chart, report, or form. Executing the stored procedure creates the HTML code used to display the component.

container folder

A repository for related items including files, text, documents, images, application components, and so on. Contrast to *PL/SQL*, *URL*, and *search* folders.

content area

Collection of related information used to store content in Oracle Portal. A content area includes all the tools required to manage its content. Known as a *site* in WebDB 2.x and Oracle Portal 3.0 beta. Content areas contain a hierarchy of folders for organizing content.

Content Area Map

Displays the hierarchical organization of all folders in an Oracle Portal content area; allows users to access folders. The Content Area Map is tailored for each authorized user; only the folders you are authorized to view and/or edit are displayed.

content area administrator

User who has full privileges over the entire content area. Content area administrators set up and maintain the content area; designate folder owners and style administrators; create perspectives and categories; and manage users. Content

area administrators can also view all the folders on the content area, including any user's personal folder.

content area log

Table of all events and related information gathered as users interact with a content area. Used to generate statistics in the form of reports and graphs.

content area page

Page that establishes a common look and feel for every folder in a content area. May include such items as a standard navigation bar, a corporate logo, and so on. Controlled by the content area administrator.

content contributor

User who can add items to a content area folder; that is, one who has Own Item, Manage Item, View Item, or Create With Approval privileges. Item and folder owners control which users can add items to a given folder.

cookie

Used by the Login Server to store information about the identity of users attempting to log in to Oracle Portal or associated external and partner applications.

custom folder type

See *folder type*.

custom item type

See *item type*.

customization form

Page that prompts end users for values to pass to an Oracle Portal component. End users can view the customization form for a component, if one has been created, by clicking the Customization link.

dashboard

Set of buttons that appears at the top of the Category, Perspective, and Folder pages when displayed in Edit mode. Allows quick access to all the edit tools used to create and manage categories, perspectives, or folders.

database access descriptor (DAD)

Set of values that specifies how Oracle Portal connects to the Apache Listener to fulfill an HTTP request. The information in the DAD includes the user name (which

also specifies the schema and the privileges), the user password, and connect string parameters.

database administrator (DBA)

Oracle Portal user belonging to the DBA group. By default, members in the DBA group have access to all Oracle Portal product pages, and have the Manage privilege for pages, content areas, applications, and administration.

database object

See *object*.

database user account

See *schema*.

direct access URLs

Allows users to bypass standard Oracle Portal navigation and directly access pages, folders, categories, perspectives, and documents using a URL.

directory

See *portal directory*.

display name

An object's external name used throughout Oracle Portal: in the Navigator, as well as on the page, folder, or component banner (unless it has been changed). When the object is published as a portlet, the display name is used as the title of the portlet in the Available Portlets list and in the portlet repository.

Edit mode

Mode that enables an authorized user who has logged on to a content area to manage folders and the items contained within the folders. To switch to Edit mode, the user clicks the Edit button at the top right of the folder page. If the Edit button does not appear, the user is not authorized to add items to the folder and/or to change the folder's properties.

expiration period

Number of days--or an exact date--on which an item expires. After an item expires, it is viewable only by the item's owner and the content area administrator. Expired items are removed from the database during a system purge of all expired items.

export

Moves source data from an Oracle Portal schema to a target schema for later retrieval. For example, you may want to export data from your production database and import all or part of the data in that export file into your development database.

external application

Application external to Oracle Portal that is launched from the External Applications portlet. If you have registered your user name and password for an application with this portlet, the Login Server logs on to the application for you each time you launch it.

favorite

Item in the Favorites portlet that provides quick access to a frequently visited URL, either within Oracle Portal or externally. Each user who logs on to Oracle Portal has his or her own Favorites portlet.

file item

One of the item types you can add to a folder. When you add a file item, the file is uploaded to the content area and stored in the host database. When a user clicks the display name link, the file is downloaded to the user's computer and displayed in the user's Web browser.

folder

Repository for content within a content area. Oracle Portal supports four folder types: container, PL/SQL, URL, and search. Folders in a content area are similar to folders on a desktop computer in that they are organized hierarchically. Folders may be accessed through the Navigator and the Content Area Map.

folder link item

One of the item types you can add to a folder. A folder link provides a route to another folder within a content area. When the user clicks the display name link, the folder referenced by the item is displayed in the user's browser.

folder owner

User responsible for managing the content (items) in a folder. May add, edit, or delete any item in the folder.

folder area

Area on the folder page where content items are displayed.

folder path

Chain of folder reference names separated by colons, starting at the root folder and ending at the object folder. Folder paths, which may appear on the folder page, describe the complete directory path of a folder and create URLs for use in folder links.

folder type

Defines the content of a folder and the information that is stored about a folder. Base folder types included with Oracle Portal are: container, PL/SQL, search, and URL. Custom folder types are folder types created by content area administrators to extend the functionality provided by base folder types, and store additional information about folders.

form

Component that provides an interface to one or more database tables, views, or procedures.

frame driver

Component consisting of a Web page divided into two frames. One frame (the driving frame) contains a SQL query that drives the contents of the second (target) frame.

function

PL/SQL subprogram that performs a specified sequence of actions and then returns a value. Functions are usually small, very specific blocks of code written to perform a specific task within the scope of a larger application.

End users can execute procedures or functions in a content area by clicking on the title of a PL/SQL or custom item.

generate

Saving a procedure containing an Oracle Portal component to a file or database in binary format so that it can be executed in run-time and batch mode.

gist

A generic *interMedia* Text gist is a summary consisting of the document paragraphs which best represent the overall subject matter. You can use the Generic Gist to skim the main content of the text, or assess your interest in the text's subject matter.

grantee

User who is given privileges on an object by another user.

group

Collection of Oracle Portal users that typically shares a common need or interest; for example, Human Resources, Accounting, and so on. Groups make it easy to grant access to an object (such as a page or component) to several users at once.

group owner

User who has the privilege to add or delete members from the group, or to delete the group itself. Each group can have more than one owner.

hierarchy

Component that displays data from a self-referencing table or view (at least two columns in the table must share a recursive relationship). A hierarchy can contain up to three levels and display data such as employees in an organization chart, or the hierarchical relationship between menus in a Web site.

home page

Within Oracle Portal, the page that is displayed upon logging on. The portal administrator chooses this page for public users; authorized users may choose their own.

HTML

Acronym for Hypertext Markup Language. A tag-based ASCII language used to specify the content, format, and links to other pages on Web servers on the Internet. Oracle Portal consists of a collection of PL/SQL procedures that, when executed, generate HTML.

HTTP function

See *function*.

image

Bitmap object that can be stored and loaded into a component or content area, and displayed using a Web browser. Oracle Portal supports images of type JPEG, GIF, or PNG.

image/imagemap item

One of the item types you can add to a folder. You can add a bitmap image or an imagemap item (collection of smaller images, each of which provides a link to a different URL).

index

Optional structure associated with a table used to locate rows of the table quickly, and (optionally) to guarantee that every row is unique.

interest list

Folder link(s) that an authorized user chooses to display in the Favorites portlet for quick and easy access.

***interMedia* Text**

Advanced gist, theme, and text retrieval services offered in Oracle 8i that enable search criteria to be compared against every word in every item within a content area. When *interMedia* is not available, or has not been enabled, search criteria is compared to each titles, descriptions, and keywords only.

item

Basic unit of content (such as text, files, and folder links) placed in a content area folder. Items are stored in the content area's host database. The item's display name and its content (or results of execution, in the case of a link or function call) are displayed on automatically-generated Web pages by Oracle Portal. Each item must be defined as one of the base item types (or custom item types, if they have been created), so that Oracle Portal knows how to display and execute it.

item ID

Local database reference to the contents of an item. An item ID value is used in custom item types to pass items to PL/SQL procedures. The function uses the item ID to access the contents of the item.

item level security (ILS)

Mechanism which allows controlled and granular access to specific items in a given folder. ILS authorizes item owners to grant explicit item access to user(s) and group(s) who would otherwise not be authorized.

item type

Define the contents of an item and the information that is stored about an item. Base item types included with Oracle Portal are: File, Folder Link, Image, Java

Application, PL/SQL, Text, URL, Application Component, and Zip File. Custom item types are item types created by content area administrators to extend the functionality provided by base item types, and store additional information about items.

item version control

See *version control*.

JavaScript

Scripting language developed by Netscape that allows generation of dynamic components in otherwise static HTML. Oracle Portal allows you to use JavaScript to create routines that validate entry fields in forms and customization forms. You can also create JavaScript event handlers for entry fields and buttons on forms.

library

Collection of one or more PL/SQL program units stored together in a database, and can be referenced by several applications at once.

link

Component that allows developers to add hypertext jumps between components.

link text

Text (usually displayed in color) which is "hot", or hypertext. End users click on the text to display the Web page specified in the HTML code. For example, in the HTML code `Travel The World`, 'Travel The World' is the link text displayed in color. End users click it to jump to the www.traveltheworld.com Web site.

List of Values (LOV)

Component that allows developers to add selectable values to entry fields in components and customization forms. An end user selects from the list one or more values for the entry field. A single List of Values can be displayed in different formats such as combo boxes, radio buttons, or check boxes.

local node

Instance of Oracle Portal to which a user is logged on. Each node has its own portlet repository that can provide portlets for other instances of Oracle Portal. Contrast with *remote node*.

lock

Setting automatically applied to a component when it is being edited. The setting prevents other users from editing the component.

Login Server

When set up to run with Oracle Portal, provides a Single Sign-On (SSO) mechanism that enables an Oracle Portal user to log in securely to Oracle Portal, and any partner and external applications, using a single user name and password.

master-detail form

Component that displays a master table row and multiple detail rows within a single HTML page. Values in the master row determine which detail rows are displayed for querying, updating, inserting, and deleting.

menu

Component that displays a Web page containing options that end users can click to navigate to other menus, Oracle Portal components, or URLs.

mod_plsql gateway

Provides support for building and deploying PL/SQL-based applications on the Web, such as Oracle Portal. PL/SQL procedures stored in the database can retrieve data from database tables and generate HTTP responses containing formatted data and HTML code to display in a Web browser.

navigation bar

In a content area, the area that allows users to navigate to frequently accessed or important areas. In framed browsers, the navigation bar appears on the left side of each folder page; in unframed browsers, it appears at the top of the page. While folder owners decide which text links or images to place on the navigation bar for a given folder, the content area administrator has control over certain elements of the navigation bar, such as which image is used as the site logo, and the text of the links themselves.

Navigator

Mechanism for locating and interacting with Oracle Portal objects, automatically customized for each user's authorization level. Provides a Find field for locating objects quickly.

object

- 1) An Oracle Portal structure used to store data in the database. Developers can create objects such as folders, components, pages, styles, and so on using wizards provided by Oracle Portal.
- 2) Database objects such as tables, procedures, Java objects, etc. These objects can also be created using Oracle Portal wizards or Oracle database commands.

Oracle connect string

Gateway setting that can be used to set up a TNS names alias for a remote database installed on Windows NT.

Oracle home

Environment variable that indicates the root directory of Oracle products.

package

Database object consisting of a specification and a body. The specification includes the datatypes and subprograms that can be referenced by other program units. The body includes the actual implementation of the package.

page

Object created in Oracle Portal containing *portlets*. Each time you display a page, it is dynamically assembled and formatted according to the portlets and layout chosen for that page.

page layout

Defines how many regions appear on a page and how they are organized.

page style

See *style*.

parameter entry field

Field on a customization form that allows end users to enter values that will be passed to an Oracle Portal component.

parent-child relationship

See *recursive relationship*.

personal folder

Area within Oracle Portal in which authorized users can store and share personal items. The portal administrator can choose to create a personal folder for a user when creating a Single Sign-On user account.

perspective

Cross-category grouping of items. Perspectives help users answer the question "Who will be interested in this item?". For example, you can add links to diverse vacation spots around the world and assign perspectives like 'Vacations for Nordic Enthusiasts', 'Archeology Expeditions', and 'Extreme Vacations for Adventurers'. Perspectives are optional, and more than one can be assigned to an item.

PL/SQL item

One of the item types you can add to a folder. A PL/SQL item contains a block of PL/SQL code. When a user clicks the item's display name link, the block is executed. The result is returned and displayed in the user's browser.

PL/SQL folder

One of the four folder types supported by Oracle Portal. PL/SQL folders contain PL/SQL code which generates HTML when the folder is rendered. Contrast with *container*, *search*, and *URL* folders.

PL/SQL function

See *function*.

portal

Set of information--content areas, pages, applications, even data from outside sources--brought together in one central location and accessed through a common interface, called a *page*.

portal administrator

Outside of the DBA, users with the highest level of privileges in Oracle Portal. Portal administrators can view and modify anything in Oracle Portal, even folders, pages, and applications marked private. (The only exception is groups: although portal administrators can modify the PORTAL_ADMINISTRATORS and PORTAL_PUBLISHERS groups, they cannot modify any other group unless they have been named group owner.)

portal directory

Lists information about Oracle Portal users, which other users can access. The portal administrator decides whether a user is included in the directory. Each user can specify whether to include personal information, such as home contact information, in the directory.

portlet

Re-usable information component that summarizes or provides access to an information source. Portlets are the fundamental building blocks of an Oracle Portal page.

portlet provider

Product (for example, iSyndicate or Oracle Portal itself) that manages portlets and provides special services, such as stock quotes, local news, and weather. Portlet providers provide code to registered partners that, when executed, downloads content to the partner's portal page. Partners can embed this code in their HTML to create content for an Oracle Portal portlet.

portlet publisher

User who can publish objects (folders, pages, components, and so on) as portlets so they can be included on pages.

portlet repository

Content area created by Oracle Portal containing all the portlets available on the local node and any registered remote nodes. When you register a new provider on your local node, or when you register remote nodes and perform a refresh, the providers and their portlets are automatically added to your portlet repository. In addition, as soon as an object in a content area is published to the portal, that content area (and the object) appear in the repository.

primary key

Column in a database table consisting of unique values that can be used to identify rows in a table.

privilege

The right to perform an action within Oracle Portal. Privileges are either global (set through the Users or Groups portlets) or specific to particular objects (usually set through the object's Access tab). When building applications, access also can be granted to database objects, shared components, components and applications.

procedure

PL/SQL subprogram that performs a specified sequence of actions.

profile

System and Oracle database resources that are available to the user.

property sheet

Displays item attributes or folder properties without having to access the Item Manager or Folder Manager.

provider

See *portlet provider*.

provider record

Record returned by a portlet provider containing specified information about a portlet.

proxy server

Server that provides access to files from other servers by retrieving them either from the local cache or from the remote server.

public folder

Any folder in a content area that is viewable by public users, who do not log on. The folder owner or content area administrator must explicitly designate a folder as public.

public user

User who can access but is not logged on to Oracle Portal. When users first access Oracle Portal they do so as public users, whether or not they have the ability to log on. A public user can view any content area folder that has been marked as public, but cannot edit any content.

Query by Example (QBE) report

Component that provides an interface allowing end users to query or insert values into a database table or view. The Query by Example report contains entry fields that correspond to the columns in the database table or view on which the form is built.

query

SQL SELECT statement that specifies which data to retrieve from one or more tables or views in a database.

partner application

Application that has delegated its authentication to the Login Server. If registered with the Login Server, users can log into a partner applications using the Single Sign-On mechanism.

purge

See *system purge*.

recent object

Oracle Portlet object recently displayed or edited. Each user has his or her own Recent Objects portlet, which provides links to the last *n* objects accessed.

recursive relationship

Relationship that occurs when the values in a table column can be related to those in another column in the same table or another table; for example, between a primary key and foreign key.

reference path

Path that uniquely identifies a portlet instance on a page. A reference path can be used to target parameters to individual portlets, or groups of portlets, on a given page.

region

Area within a page. Each region has its own specifications regarding how portlets placed within it are displayed (as rows or columns), how wide the region is relative to other regions in the row, and so on.

remote database

Database running on a separate machine that can be accessed over the network through a connect string or database link.

remote node

Any instance of Oracle Portal other than the one you are currently logged on to (known as the *local node*).

report

Oracle Portal component that displays the results of a SQL query in a tabular format.

role

Group of Oracle database users to whom database object privileges can be granted and revoked as a unit. A Database Administrator assigns database users roles in order to grant them Oracle database object privileges associated with the role.

rollover image

Second image that is displayed whenever the cursor is moved over an image on the navigation bar in a Oracle Portal content area, or on a tab on an Oracle Portal page.

root folder

Top level of the folder hierarchy in a content area; contains all other folders in the content area. Also known as the content area's home page.

row

Set of values in a table; for example, the values representing one employee in the SCOTT.EMP table.

schema

Collection of components and database objects under the control of a given database user.

search folder

Folder based on an Oracle Portal content area search which creates different views of the content for different audiences. For example, you can create a search folder for all items belonging to the same category. Search folders are dynamic and are updated each time the folder is rendered. Contrast to *container*, *PL/SQL* and *URL* folders.

sequence

Database object used to automatically generate numbers for table rows.

session

Uninterrupted connection between a browser and Oracle Portal, from initial access to log off or disconnect. A session established as a **public session** at initial access becomes an **authenticated session** upon login.

shared components

Building blocks used by Oracle Portal developers to create components. Shared components include JavaScripts, color, font, and image definitions. Each shared component can be used by multiple developers to create Oracle Portal components.

shared objects

Content area created by Oracle Portal which contains objects such as personal folders, shared perspectives, shared categories, and shared custom types (shared attributes, shared item types and shared folder types). These objects can be shared across all content areas.

Single Sign-On

See *Login Server*.

site

See *content area*.

snapshot

Table that contains the results of a query on one or more tables, called master tables, in a remote database.

snapshot log

Table associated with the master table of a snapshot tracking changes to the master table.

status code

See *version*.

stored procedure

See *procedure*.

structured U/I template

Shared component that controls the look and feel of Oracle Portal components. Structured U/I templates display the same image and text in the same location on every component that uses the template.

style

Set of values and parameters that controls the appearance of Oracle Portal folders and pages. Styles applied to folders can control the look and feel of all the folders in

a content area, or they can be applied to a single folder. Page styles are always applied on a page-by-page basis. Styles created by the Style Manager in the content area can be applied to folders, banners, and navigation bars; page styles are applied only to pages.

style administrator

User who has special privileges to maintain and enforce display settings in a given page or folder style.

subfolder

A nested folder. All folders in a content area are subfolders of the content area's home page, also known as the root folder.

substitution tag

Tag used to create unstructured U/I templates. When the HTML code that creates the template executes, substitution tags dynamically embed components, titles, headings, and other elements into the template.

synonym

Name assigned to a table or view that can thereafter be used to refer to it.

system purge

Deletes all items in a content area marked as deleted or expired from the database. System purges are performed by the content area or portal administrator.

table

Basic storage structure in a relational database.

tablespace

Allocation of space in the database that can contain objects.

temporary tablespace

Allocation of space in the database used for the creation of temporary table segments for operations such as sorting table rows.

text item

One of the item types you can add to a folder. When you create a text item, you enter text (up to 32KB) in the Item Wizard. The text block is then stored in the content area's host database. When a user clicks the display name link, the text is displayed in the user's Web browser.

theme

An *interMedia* text theme is a snapshot that describes what the document is about. Rather than searching for documents that contain specific words or phrases, users can search for documents that are about a certain subject, even if that subject is not mentioned explicitly in the document. Theme queries return a list of documents that are about the requested subject, along with a score that indicates how strongly each document refers to the subject in question.

title

See *display name*.

translation

Content area rendered in another language. When a content area administrator creates a translation, content contributors can add content in that language. Content area users can also view the content by setting their browser language appropriately.

trigger

Database object associated with a table. It executes before or after one or more specified events.

unstructured U/I template

Shared component that controls the look and feel of Oracle Portal components. Unstructured U/I templates are based on HTML code that, when executed, dynamically embeds components, titles, headings, and other elements.

URL folder

Folder providing a route to another Web page, either inside or outside this content area. When a user clicks the folder link, the Web page referenced by the link is displayed. Contrast to *container*, *PL/SQL*, and *search* folders.

URL item

One of the item types you can add to a folder. URL items provides a route to another Web page, either inside or outside this content area. When a user clicks the item's display name link, the Web page referenced by the link is displayed.

user interface (U/I) template

Shared component that controls the look and feel of Oracle Portal components. Selecting a U/I template when building a component automatically selects a title on

the page where the component is displayed, a title background, links to other web pages, and background colors and images.

version

Indicates the status of a stored procedure that contains a Oracle Portal component. For example, ARCHIVE indicates an old version of the component that is being saved in the database. PRODUCTION with VALID PACKAGE indicates the most recent version of the component, which will run without errors. PRODUCTION with INVALID PACKAGE indicates that the most recent version of the component contains errors. There can be multiple versions of the same component.

version control

Allows multiple versions of an item to simultaneously exist in the Oracle database. This feature is useful for tracking document changes from one version to the next or for reverting to a previous version if necessary.

view

A virtual table whose rows do not actually exist in the database, but which is based on a table that is physically stored in the database.

View mode

Usual mode of viewing and interacting with an Oracle Portal content area. Contrast to *Edit mode*, which is used to perform folder and item management tasks.

visited link

Link which is displayed in a different color than unvisited links to signify that the user has clicked it at least once during the browser session.

Web server

Program that delivers Web pages.

wildcard

In Oracle Portal, the percent (%) character, which is used to mean any single character or a contiguous set of characters within a word or phrase.

wizard

Graphical interface that guides a user step by step through a process. In Oracle Portal, there are wizards for creating applications, components, database objects pages, content areas, folders, and items within folders.

Zip file item

Method for uploading many files in one operation. May be used to replicate a file system or Web site. A folder is created for each directory and an item is created for each file. The items are contained in a target folder in the chosen content area.

Index

A

access

- control access to components, 5-34
- granting database object privileges, 3-15
- granting folder privileges, 6-23

Access Control List

- definition, 1-7

ACL

- see Access Control List, 1-7

adding as a portlet, 4-38

application

- definition, 1-6

attribute

- by item type (optional), 2-21
- by item type (required), 2-20
- creating, 6-10

B

bind variable

- creating dynamic list of values, 4-23

bookmark

- adding as a portlet, 3-5
- adding to a portlet, 3-6
- removing from a portlet, 3-7

button

- creating using PL/SQL, 3-17

C

calendar

- add as a portlet, 3-18
- create button to access form, 3-17

create to display events, 3-8

create to display vacation request dates, 5-31

creating, 3-7, 3-8

using a form to update, 3-10

Candidate Category Structure Worksheet, 2-15

category

assigning to an item, 6-6

Candidate Category Structure Worksheet, 2-15

creating, 6-5

definition, 1-5

for content area, 2-13

chart

add as a portlet, 4-12

adding multiple to a single portlet, 4-15

create, 4-11

creating, 4-17

testing, 4-12

component

add as a portlet, 3-18

adding as a portlet, 4-12

adding multiple to a single portlet, 4-15

adding to a portlet, 4-38

adding various to a single portlet, 4-27

create button to access another component, 3-17

creating a chart, 4-11, 4-17

creating a report, 4-18

creating list values, 4-21

displaying two or more in one portlet, 4-14, 4-16

linking, 3-16, 4-10

content area

difference from pages, 1-6

folder structure, 2-11

implementing security, 2-12

installing the demo, A-4

- Item Worksheet, 2-8
 - overview, 1-5
 - planning, 2-6
 - troubleshooting, 2-24
- User Community Worksheet, 2-9
- content area style
 - definition, 1-5
- Creating, A-1
- custom item type
 - adding a procedure call, 6-14
 - creating, 6-10, 6-12
 - creating an attribute, 6-10

D

- database object
 - creating a database sequence, 3-14, 4-28
- database sequence
 - create, 5-17
 - create for a form, 5-26
 - creating, 3-14, 4-28
- demo
 - installing portals, schema, and content areas, A-4
- display
 - control display of components in portal, 5-34

E

- Entertainment Content Area
 - folder structure, 2-6
- event form
 - creating, 3-10
- External Applications, 2-3

F

- favorite
 - adding a link, 3-6
 - removing a link, 3-7
- Favorites portlet
 - adding, 3-5
 - adding a link, 3-6
 - removing a link, 3-7
- folder

- adding as a portlet, 4-5
- changing layout, 6-17
- changing region properties, 6-18
- Folder Tree Worksheet, 2-18
- form
 - changing appearance, 3-12
 - create a database sequence for a form, 5-17
 - create to look up a job vacancy, 5-15
 - create to post or close a job vacancy, 5-17
 - create to post or close job vacancies, 5-19
 - create to request a vacation, 5-26, 5-27
 - creating, 3-10, 4-24, 4-28
 - creating a database sequence to update records, 3-14
 - creating to update events in a calendar, 3-10
- linking to a hierarchy, 5-12
- using with LOVs, 4-24

G

- group
 - creating, A-3

H

- hierarchy
 - creating, 5-8
- HTML
 - adding to a portlet, 3-4
- HTML portlet, 2-3
 - adding, 3-4

I

- item
 - adding to a content area, 6-13, 6-16
 - adding to a region, 6-21
 - assigning a category, 6-6
 - assigning a perspective, 6-9
 - attribute, 1-5
 - characterizing, 1-5
 - moving to a different region, 6-22
 - optional attributes for, 2-21
 - required attributes for, 2-20
 - zip file, 6-2

item type

- adding a procedure call, 6-14
- creating, 6-10, 6-12
- creating an attribute, 6-10

Item Worksheet, 2-8

L

layout, 3-3

- changing folder layout, 6-17
- changing region properties, 6-18
- create, 4-3

link

- create between a report and a form, 5-23
- create between components, 5-23
- create between report and form, 5-23
- creating between a form and hierarchy, 5-12
- creating between components, 3-16, 4-10
- linking a chart to a report, 4-6
- linking a report to a chart, 4-6
- testing chart with links, 4-12

list of values

- create, 5-18
- create a pop-up list, 5-26
- create an employee list of values, 5-26
- creating, 4-21
- creating dynamic, 4-21, 4-22
- creating dynamic with bind variable, 4-23
- creating LOVs dependant on each other, 4-23
- creating static, 4-22

local node, definition, 1-4

LOVs

- see list of values, 4-21

M

menu, 4-38

multiple files

- adding to a content area, 6-2

N

node

- definition, 1-4

O

Oracle Portal

- overview, 1-1

organization chart

- creating, 5-8
- linking from a form, 5-12

P

page

- adding as a portlet, 4-3
- create tabs, 4-3
- creating, 3-3
- definition, 1-3
- difference from content areas, 1-6
- enabling to be added as a portlet, 4-3
- planning, 2-1

page region

- definition, 1-3

page style

- creating, 2-4
- definition, 1-3

perspective

- assigning to an item, 6-9
- creating, 6-7
- definition, 1-6
- for content area, 2-14
- icon, 6-8

portal

- about, 3-1
- adding a menu, 4-38
- adding a portlet, 3-18
- control display of components, 5-34
- creating, 3-1, 4-1, 5-1
- creating a simple intranet portal, 3-1
- creating for a line of business, 4-1
- creating for Human Resources department, 5-1
- creating for intranet, 3-1, 4-1, 5-1
- definition, 1-3
- design HR portal tabs, 5-37
- displaying to the public, 3-19
- installing the demo, A-4
- using security, 5-1

portal administrator

- creating a group, A-3
- creating a user, A-1
- installing demo portals, schema, and content areas, A-4
- portlet
 - add to a page, 5-37, 5-39, 5-40
 - adding, 3-18
 - adding a chart, 4-12
 - adding a folder, 4-5
 - adding a menu, 4-38
 - adding a page as a portlet, 4-3
 - adding multiple components, 4-27
 - creating multiple tabs, 4-14, 4-15, 4-16, 4-27
 - definition, 1-3
 - displaying various components in one portlet, 4-16
- portlet provider
 - definition, 1-4
- portlets
 - arrange on a page, 5-37
 - content area objects, 6-25
- preface
 - conventions table sample, xvi
- privileges
 - creating the demo portals, 3-1, 4-1, 5-4
 - granting folder access, 6-23

R

- region
 - adding an item, 6-21
 - changing layout, 6-17
 - changing properties, 6-18
 - moving an item, 6-22
- remote node
 - definition, 1-4
- report
 - create from SQL query, 5-24, 5-32
 - create to display current job vacancies, 5-24
 - create to display job vacancies, 5-24
 - create to display vacation balances, 5-32
 - creating, 4-6, 4-18
 - testing, 4-9

S

- schema
 - installing the demo, A-4
- security, 1-7
 - apply to components, 5-34
 - for content area, 2-12
 - for pages, 2-5
 - granting component access to specific groups, 5-34
- style, 3-3

T

- tab
 - adding portlets, 5-37, 5-39, 5-40
 - creating in portlets, 4-14, 4-16, 4-27
 - creating tabbed pages, 4-3, 5-5
 - design look and feel, 5-37
 - displaying to a group, 5-6, 5-7
 - using security to display or hide, 5-5

U

- user
 - creating, A-1
- User Community Worksheet, 2-9
 - sample, 2-11

W

- wizard
 - Create Calendar, 3-8, 5-31
 - Create Chart from Query Wizard, 4-11
 - Create Chart from SQL Query Wizard, 4-17
 - Create Database Sequence, 3-14, 4-28, 5-17, 5-26
 - Create Dynamic List of Values, 5-18, 5-26
 - Create Form based on Table or View, 3-10, 4-24, 5-15, 5-17, 5-19, 5-27
 - Create Hierarchy, 5-8
 - Create Link, 3-16, 4-10, 5-23
 - Create Page, 3-3, 4-3
 - Create Report from SQL Query, 5-24, 5-32
 - Create Reports from Query Wizard, 4-6, 4-18
 - Create Static List of Values, 5-18

Z

zip file item

adding to a content area, 6-2

