Oracle® Reports

Tutorial

10*g* (9.0.4)

Part Number B10612-01

September 2003

ORACLE®

Oracle Reports Tutorial, 10*g* (9.0.4)

Part Number B10612-01

Copyright © 2002, 2003 Oracle Corporation. All rights reserved.

Primary Author: Vanessa Wang

Contributors: Ellen Gravina, Reena John, Darren McBurney, and Philipp Weckerle

Contents

Glossary

Index

Send Us Your Comments

Oracle Reports Tutorial, 10*g* (9.0.4)

Part Number B10612-01

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 suggestions for improvement to this documentation, please send us your comments via the Oracle Reports discussion group forum:

`(http://otn.oracle.com/products/reports/)`

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

Welcome

Welcome to the Oracle Reports Tutorial. This manual will help you get started using Oracle Reports, as well as introduce you to publishing data to the Web and paper.

> **Note:**   For the latest updates to the *Oracle Reports Tutorial*, refer to the Oracle Technology Network (`http://otn.oracle.com/products/reports/`), then click **Getting Started with Oracle Reports** and use the index to navigate to the *Oracle Reports Tutorial*.

This preface includes the following sections

- Intended Audience
- Documentation Accessibility
- Prerequisites
- Structure

Intended Audience

This manual is intended for those users who are new to Oracle Reports, as well as those who are familiar with previous versions, but would like to learn more about some of the major new features in Oracle Reports 10*g* (9.0.4).

Documentation Accessibility

Our goal is to make Oracle products, services, and supporting documentation accessible, with good usability, to the disabled community. To that end, our documentation includes features that make information available to users of assistive technology. This documentation is available in HTML format, and contains markup to facilitate access by the disabled community. Standards will continue to evolve over time, and Oracle Corporation is actively engaged with other market-leading technology vendors to address technical obstacles so that our documentation can be accessible to all of our customers. For additional information, visit the Oracle Accessibility Program Web site at `http://www.oracle.com/accessibility/`.

Accessibility of Code Examples in Documentation JAWS, a Windows screen reader, may not always correctly read the code examples in this document. The conventions for writing code require that closing braces should appear on an otherwise empty line; however, JAWS may not always read a line of text that consists solely of a bracket or brace.

Accessibility of Links to External Web Sites in Documentation This documentation may contain links to Web sites of other companies or organizations that Oracle Corporation does not own or control. Oracle Corporation neither evaluates nor makes any representations regarding the accessibility of these Web sites. '

Prerequisites

The exercises in the tutorial use the Human Resources sample schema provided with the Oracle9*i* database and an HTML template. We've also provided a text file containing the SQL you will enter, as well as the JSPs you create in every chapter.

You can download these files from the Oracle Technology Network (`http://otn.oracle.com/products/reports`), then click **Getting Started with Oracle Reports**. Navigate to the index, then click **Examples**. On the Examples page, you will see a link to the *Oracle Reports Tutorial*. You can also find the files on the Documentation CD that came with the product. We recommend that you copy the files into a local directory, such as `d:\Reports_Tutorial`.

Sample Schema

This tutorial relies on the data contained in the Human Resources section of the sample schema. This sample schema is provided with the Oracle9*i* database.

Viewing Web Reports

To produce the Web-based report, you must have a Web browser installed on your machine. The minimum and recommended requirements are:

- Microsoft Internet Explorer 4.x or higher

or

- Netscape Communicator 4.x or higher

Viewing the Web Source

Although you can view the Web source for your JSP report in Reports Builder, this tutorial also shows you how to analyze your Web source in a text editor. We recommend you use a text editor like NotePad (on Windows NT) or UltraEdit.

Structure

This manual contains the following chapters.

Chapter 1, "Tutorial Overview"

This chapter describes the tutorial scenario and what the chapters will help you achieve.

Lesson Chapters

The lesson chapters contain the procedures for producing the reports. These chapters step you through the Report Wizard, adding report blocks, and the Graph Wizard. If you are already familiar with using these features, you can use the Quick Reference guide, located in Appendix A.

Output Review Chapters

The output review chapters review what you did in the lesson chapters. These chapters explain the JSP tags and code for the JSP-based Web report.

1

Tutorial Overview

During this tutorial, suppose you are a developer for a company called My Company. You have been asked to publish some content on the company intranet so human resources managers for each branch can view this data from any location. You must use the company's template (which is an HTML file) to make the data look good on the Web. But, you must also enable managers to print out a paper version of the report.

In this tutorial, you will build a report for the Web using JSPs (JavaServer Pages) that displays the required information about employee salaries in each department. You will also create a graph so managers can see an overview of the data. At the end of the tutorial, we'll show you how to quickly generate a paper report based on the same data model.

The following image shows an overview of the first part of the tutorial.

Figure 1–1   Tutorial Overview: Creating the Web Report
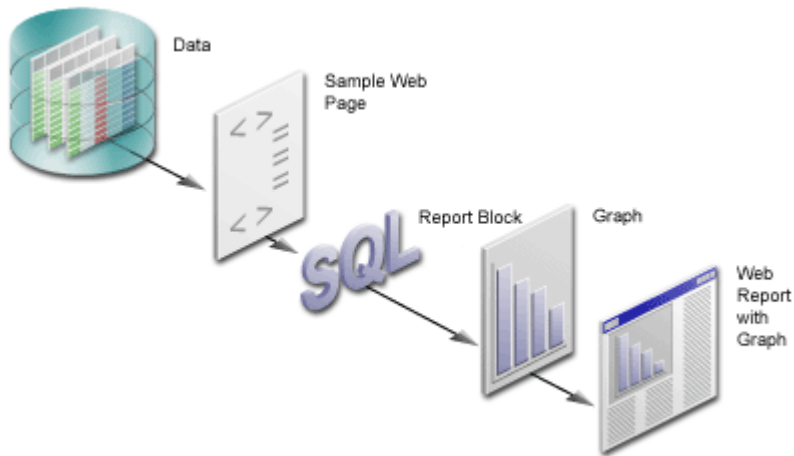
1.1 Tutorial Scenario

In the chapters of this tutorial:

1. You will open the Web page we've provided for you, which contains some simple HTML, then create a data model for the report, which will pull data from a sample data source into the report (Chapter 2, "Adding Data to a Report"). This Web page contains the template for My Company's look and feel.

Figure 1–2   Adding Data to an HTML Page
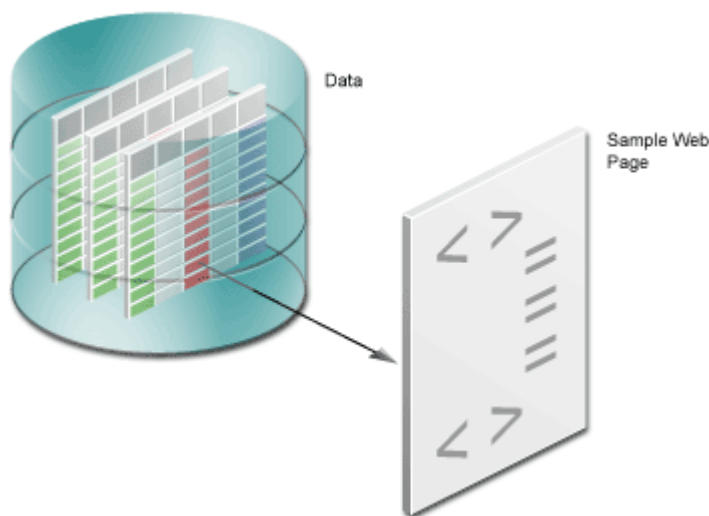
2. In Chapter 3, "Reviewing the Source Code of the Web Report", you will review the resulting report to analyze what the steps you completed in Chapter 2, "Adding Data to a Report" did to the sample Web page to help you understand what Reports Builder did to the sample Web page. You can compare the sample Web page we've provided with the resulting JSP-based Web page. Here, you will be able to examine how the data model looks in Reports Builder, and how it looks in XML.

3. In Chapter 4, "Creating a Report Block for the Web Report", you will use the Report wizard to add a report block to the JSP and generate a simple JSP-based Web report to your Web browser.

Figure 1–3   Creating a Report Block for your JSP-based Web Report
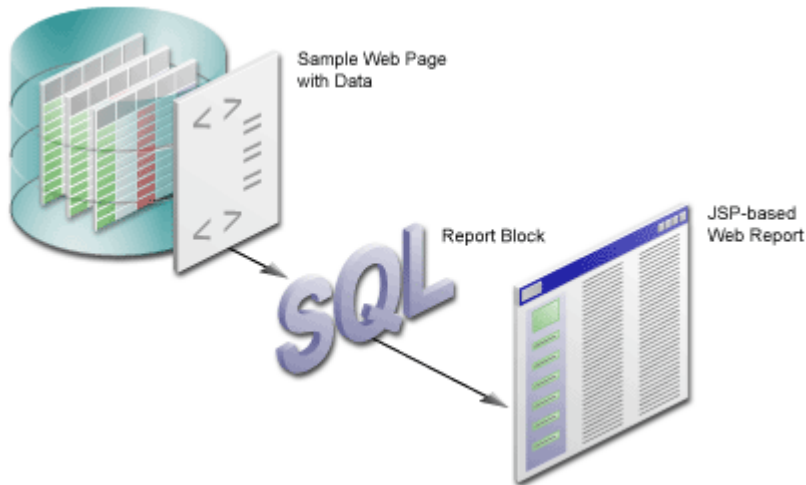
4. Then, in Chapter 5, "Reviewing the Source Code for the Report Block", you will analyze the Web source of your report to review how the report block was inserted into your JSP-based Web report. Again, you will examine the XML code to see how the source code has changed.

5. In Chapter 6, "Creating a Graph for the Web Report", you will use the Graph wizard to add a graph to the JSP, then generate the completed JSP report to your Web browser.

Figure 1–4   Adding a Graph to a JSP-based Web Report

6. In Chapter 7, "Reviewing the Source Code for the Graph", you will review the new code that Reports Builder added to your Web source to produce the graph.

7. Finally, in Chapter 8, "Generating a Paper Report", you will generate a paper report based on the same data model and layout you created for the JSP-based Web report.

Figure 1–5   Generating a Paper Report from a JSP-based Web Report Data Model
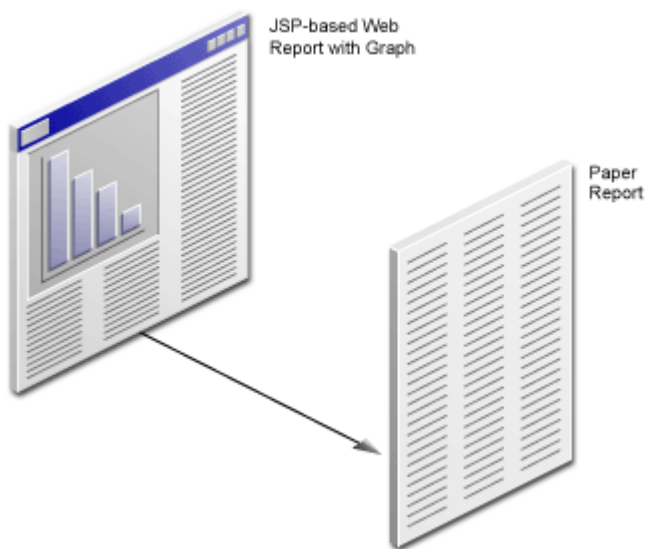
1.2 Summary

Now that you know what this tutorial covers, you can go on to Chapter 2, "Adding Data to a Report" to learn how to create a data model for your report using the Data Wizard.

2

Adding Data to a Report

Estimated completion time: 15 minutes

Suppose you need to create a Web report that displays salary information about each employee in a department for your company's human resources managers. You already have an HTML page that match the company's logo and colors, and just want to add the necessary information.

This chapter shows you how to use the Data Wizard in Reports Builder to add data to an existing HTML page. At the end of the chapter, you will have created a data model to make data available for use in a Web report.

Figure 2–1   Adding Data to an HTML Page
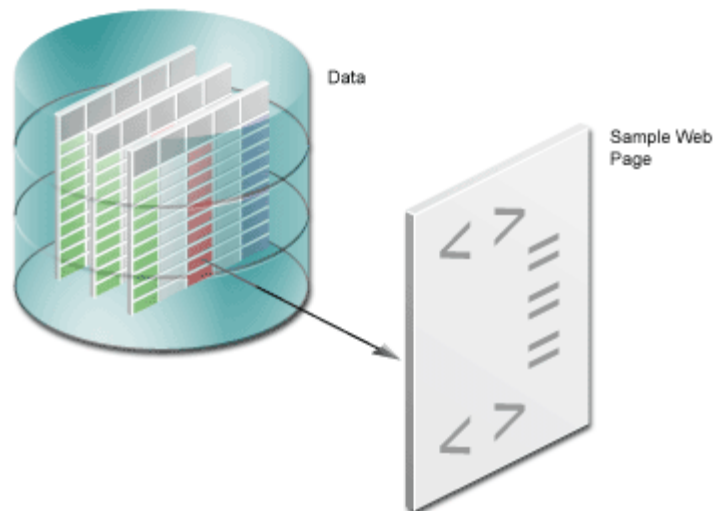
2.1 Open the Web page

The steps in this section show you how to open the Web page template we've provided, called emprev.htm. First, you will open the Web page in your Web browser so you can see what our template looks like.

> **Note:** If you do not have this file, refer to the "Prerequisites" section, in the Welcome chapter.

To open the sample HTML page in your Web browser:

■ In your Web browser (e.g., Netscape or Internet Explorer), choose **File > Open**, then navigate to the sample file we've provided, called emprev.htm.

To open an existing HTML page in Reports Builder:

1. Launch Reports Builder.

> **Note:** If you are using UNIX, navigate to the directory where Reports Builder is installed, then execute runbuilder.sh.

2. In the Welcome dialog box, click **Open an existing report,** then click **OK**.

3. In the Open dialog box, navigate to the folder where the tutorial sample files are located, such as d:\Reports_Tutorial.

4. Find the file emprev.htm, then click **Open**.

The Object Navigator now displays the emprev report.

Figure 2–2   Object Navigator displaying an existing HTML page
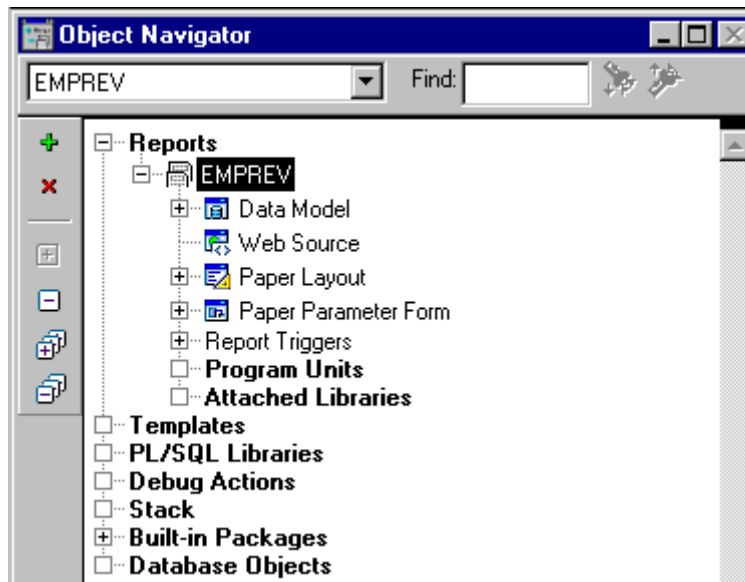
2.2 Use the Data Wizard to add data to a sample Web page

When you build a report, you must first build a data model by selecting the data that you want to use in the report.

The steps in this section show you how to use the Data Wizard to build a data model for your Web report. The data model you create makes the data from the sample schema available for you to use in your report.

To add data to an existing HTML page:

1. Open the Data Wizard to define a layout and add a data model.

   ■ In the Object Navigator, double-click the **Data Model** node.

   ■ In the Data Model view, right-click the canvas, then choose **Data Wizard** from the pop-up menu.

2. If the Welcome page displays, click **Next**.

3. On the Query name page, click **Next** to accept the default name.

4. On the Data Source page, make sure **SQL Query** is selected, then click **Next**.

> **Note:** On the Data Source page, notice the other options (Oracle9*i*, XML, JDBC, etc.). On this page, you can choose any data source you wish to use in your report. For more information about using pluggable data source, see the *Reports Builder online help* or *Getting Started with Oracle Reports*.

5. On the Data page, you can do either of the following:

   ■ Open **tutorial_sql.txt** in a text editor, copy the query, and paste it into the SQL Statement box). Then, proceed to Step 16, **OR**:

   ■ To learn how to use the Query Builder, follow steps 6 through 15.

6. To use the Query Builder, click **Query Builder**.

> **Note:** If you are not connected to a database, the Connect dialog box displays. In the Connect dialog box, type the user name and password for your database. Please note that this tutorial uses the Human Resources schema. If you are using the default login, you can use the connection string: `hr/hr@<databasename>`.
>
> If you do not know the connection information for the database that contains the sample schema we have provided, contact your administrator.

7. In the Query Builder, double-click the **EMPLOYEES** table.

8. Double-click the **EMPLOYEES** table again, then click **Close**.

Figure 2–3   Employees Tables displaying in the Query Builder

The EMPLOYEES table displays in the Query Builder as EMPLOYEES and EMPLOYEES A1. The link between the MANAGER ID in EMPLOYEES and the EMPLOYEE ID in EMPLOYEES A1 is automatically created because of the constraints that exist in the EMPLOYEES table.

> **Note:**   You'll notice that some of the column names are bold, and some are italicized. Column names that are in bold are primary keys and column names that are in italics are foreign keys.

9. In the **EMPLOYEES** table, select the check boxes for the following fields (note that you *must* select the fields in the following order):

   - EMPLOYEE_ID
   - FIRST_NAME
   - LAST_NAME
   - HIRE_DATE
   - JOB_ID
   - SALARY
   - DEPARTMENT_ID

10. In the **EMPLOYEES A1** table, select the check boxes for the following fields:

- EMPLOYEE_ID
- FIRST_NAME
- LAST_NAME

11. Click **OK**.

12. The query generated for you by Query Builder now displays in the **SQL Query Statement** text box, and should look like the following:

```
SELECT ALL EMPLOYEES.EMPLOYEE_ID
   , EMPLOYEES.FIRST_NAME,EMPLOYEES.LAST_NAME
   , EMPLOYEES.HIRE_DATE
   , EMPLOYEES.SALARY
   , EMPLOYEES.DEPARTMENT_ID
   , EMPLOYEES_A1.EMPLOYEE_ID
   , EMPLOYEES_A1.JOB_ID
   , EMPLOYEES_A1.FIRST_NAME, EMPLOYEES_A1.LAST_NAME
FROM EMPLOYEES, EMPLOYEES EMPLOYEES_A1
WHERE (EMPLOYEES.MANAGER_ID = EMPLOYEES_A1.EMPLOYEE_ID)
```

13. To restrict the retrieved data to just the employees in Department 100, add an AND clause, so that your query looks like this (new code is in bold text):

```
SELECT ALL EMPLOYEES.EMPLOYEE_ID
   , EMPLOYEES.FIRST_NAME,EMPLOYEES.LAST_NAME
   , EMPLOYEES.HIRE_DATE
   , EMPLOYEES.SALARY
   , EMPLOYEES.DEPARTMENT_ID
   , EMPLOYEES_A1.EMPLOYEE_ID
   , EMPLOYEES_A1.JOB_ID
   , EMPLOYEES_A1.FIRST_NAME, EMPLOYEES_A1.LAST_NAME
FROM EMPLOYEES, EMPLOYEES EMPLOYEES_A1
WHERE (EMPLOYEES.MANAGER_ID = EMPLOYEES_A1.EMPLOYEE_ID)
AND EMPLOYEES.DEPARTMENT_ID=100
```

14. Let's make the report look more organized by displaying the employee names with their last names first.

Concatenate the EMPLOYEES.FIRST_NAME and EMPLOYEES.LAST_NAME fields into one alias called `emp_name`.

The line of the query should now look like this:

```
, EMPLOYEES.LAST_NAME ||', '|| EMPLOYEES.FIRST_NAME emp_name
```

15. Concatenate the EMPLOYEES_A1.FIRST_NAME and EMPLOYEES_A1.LAST_NAME fields into one alias called `mgr_name`.

The line of the query should now look like this:

```
, EMPLOYEES_A1.LAST_NAME ||', '|| EMPLOYEES_A1.FIRST_NAME mgr_name
```

16. Verify that your query is correct. The entire query should now look like this:

```
SELECT ALL EMPLOYEES.EMPLOYEE_ID,
    EMPLOYEES.LAST_NAME||','|| EMPLOYEES.FIRST_NAME emp_name,
    EMPLOYEES.HIRE_DATE, EMPLOYEES.JOB_ID, EMPLOYEES.SALARY,
    EMPLOYEES.DEPARTMENT_ID, EMPLOYEES_A1.EMPLOYEE_ID,
    EMPLOYEES_A1.LAST_NAME||','|| EMPLOYEES_A1.FIRST_NAME mgr_name
    FROM EMPLOYEES, EMPLOYEES EMPLOYEES_A1
WHERE (EMPLOYEES.MANAGER_ID = EMPLOYEES_A1.EMPLOYEE_ID)
AND EMPLOYEES.DEPARTMENT_ID = 100
```

> **Note:** If you're not sure whether your query is correct, open the file **tutorial_sql.txt** from the example files we provided to you

17. Click **Next**.

18. Now that we've selected the raw data, let's organize it into groups by department manager name.

On the Groups page, in the left column, click the **MGR_NAME** field, click the right arrow to move the field to the **Group Fields** list, then click **Next**.

19. Now, let's calculate some of our data. The Totals page displays some of the commonly-used calculations.

On the Totals page, in the left column, click **SALARY**, then click **Sum** to display the sum of the Salary column in the **Totals** list.

20. Click **Finish**.

Figure 2–4   Data Model view for the report
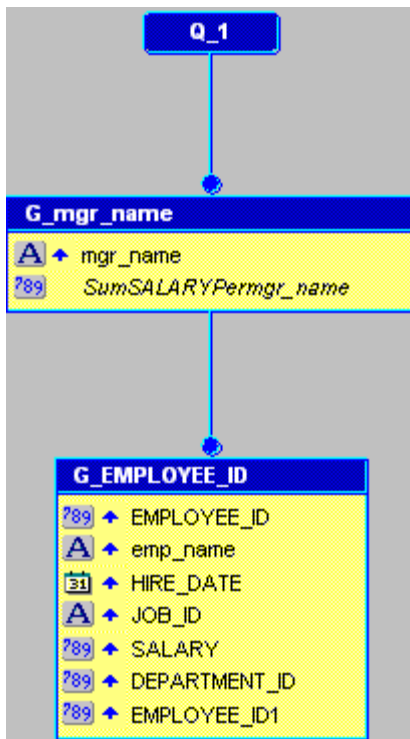
> **Note:**   A Reports-level summary displays in the upper left-hand
> section of the Data Model view, but is not displayed in this image.

2.3 Save your report as a JSP file

In Reports Builder, you can save your report using several formats. Since we're
creating a JSP-based Web report, we will save our report as a JSP (JavaServer Page).

To save your report as a JSP:

1.   With your report selected in the Object Navigator, choose **File > Save As**.

2.   In the Save dialog box, change the report name to emprev_<your
     initials>, and change the type to **Reports JSP (.jsp)**, then click **Save**.

> **Note:** Be sure to include your initials so you do not overwrite the files we've provided. This saves the report as a JavaServer Page (JSP). Since JSP is the primary technology Reports Builder uses to publish reports to the Web, saving your report as a JSP prepares your report for the later chapters in the tutorial.

3. Choose **File > Close**.

2.4 Summary

Congratulations! You have now created a data model that will allow you to include a report on your Web page. You now know how to:

- Open an existing Web page (HTML file) in Reports Builder
- Use the Data Wizard to specify data for a report
- Use the Query Builder to select data
- Save your report as a JavaServer Page (JSP)

To review your work, continue to Chapter 3, "Reviewing the Source Code of the Web Report". Otherwise, skip to Chapter 4, "Creating a Report Block for the Web Report" to continue building your report.

For more information on using the Report Wizard, columns, groups, and fields, see the *Reports Builder online help*. You can also find more information about why JavaServer Pages are useful by visiting *Getting Started with Oracle Reports* on the Oracle Technology Network (`http://otn.oracle.com/products/reports/`).

3

Reviewing the Source Code of the Web Report

Estimated completion time: 5 minutes

In Chapter 2, "Adding Data to a Report", you added a data model to your Web page. Here, we show the relationship between the entries you made in the wizard and the data model you created, as well as the JSP and XML code.

You will also examine the custom JSP tags that Reports Builder inserted into your JSP. These custom JSP tags enable Oracle Reports to add the data you've chosen in the Data Wizard to the JSP-based Web report.

3.1 View the Web source in Reports Builder

To view the Web source of your report in Reports Builder:

1. Make sure the report `emprev_<your initials>.jsp` is open in Reports Builder.

2. In the Object Navigator, double-click the **Data Model** node to display the data model for this report. Your data model should look something like this:

Figure 3–1   Data Model for your HTML report
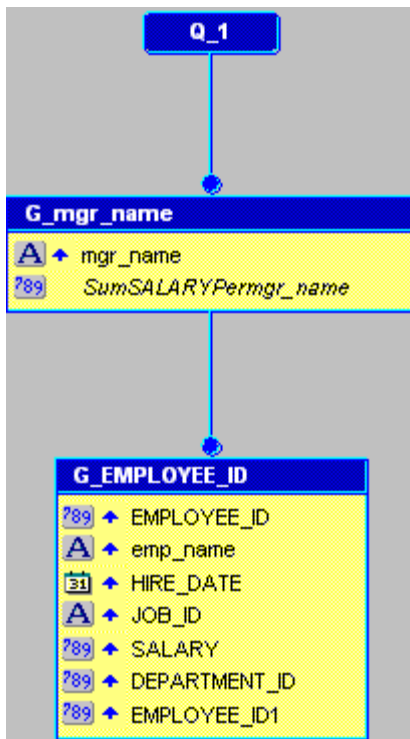
3. In the Data Model view, click the **Web Source** icon in the toolbar.

> **Note:**   You can also double-click the **Web Source** icon under the report name in the Object Navigator.

The source code displays in the Web Source view. In this code, scroll through and note the following items:

- The @ taglib line references the reports JSP library for all tags starting with rw. The `<%...%>` tags mark JSP-relevant tags, and point to the JSP Custom Tag Library.

- Notice the opening `rw:report` tag and the opening and closing `rw:objects` tags.

- The `rw:report` tag appears after the library call. In a report, all Reports Builder JSP tags must appear between an opening and closing rw:report tag. If the closing `rw:report` tag comes before the data is used, the report will be empty.

- The `rw:objects` tags appear in the Web Source. Note that you do not see any text between the opening and closing tags. When you save the report to your file system, Reports Builder encodes the data model and other elements in a language called XML, and places the data model between these tags. Reports Builder hides the XML code between these tags for the sake of maintaining your XML integrity.

> **Note:** Unless you are comfortable using XML and JSPs, we recommend that you do *not* type anything between these tags (rw:objects) in this view.

3.2 Verify your JSP code

If you've gone through this chapter and still aren't sure whether your JSP is correct, open the file we've provided, called `emprev.jsp`. The file is located in the tutorial examples directory you created.

3.3 Summary

Congratulations! You have finished reviewing the data you added to your Web report in Chapter 2, "Adding Data to a Report". You now know how to view the source code in the Reports Builder Web Source view and have learned about these Oracle Reports custom JSP tags:

- `rw:report`

- `rw:object`

Continue to Chapter 4, "Creating a Report Block for the Web Report" to add a report block and finish your Web report.

For more information on Oracle Reports JSP tags, see the *Reports Builder online help*.

4

Creating a Report Block for the Web Report

Estimated completion time: 15 minutes

Now that you've created your data set, you want to format the data into your report to provide managers in your company with current information about their employees. To do so, you can create a report block and add it to your report. This report block will dynamically pull in the data using the query you specified in Chapter 1 every time you run the report, and format the data in your report.

The steps in this chapter will show you how to add a report block to your Web page.

Figure 4–1   Creating a Report Block for your JSP-based Web Report
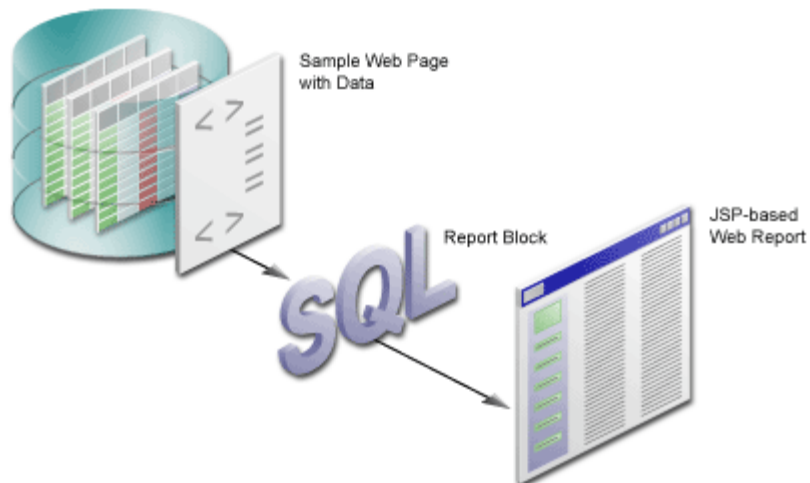
4.1 Add a report block to your Web page

If you completed Chapter 3, "Reviewing the Source Code of the Web Report", you should already have the source open in Reports Builder, and you can skip Step 1. The steps in this section show you how to use the Report Wizard to format the data you specified in Chapter 1, and dynamically pull in the data into your report.

To add a report block:

1. Make sure your report (`emprev_<your initials>.jsp`) is open in Reports Builder.

   > **Note:**   If you did not create `emprev_<your initials>` in Chapter 1, simply open the Web page we've provided, named `emprev.jsp`.

2. In the Object Navigator, double-click the Web Source node under `emprev_<your initials>`.

3. In the Web Source view, choose **Edit > Find and Replace** to find and delete the following text (note that this text is in two different places that state "Replace"):

   ```
   Replace this text with the Report block in Chapter 4 of the tutorial.
   ```

4. Make sure your cursor is in the location where you deleted the text, and choose **Insert > Report Block** to display the Report Wizard.

Figure 4–2   Inserting the Report Block into your Web Source
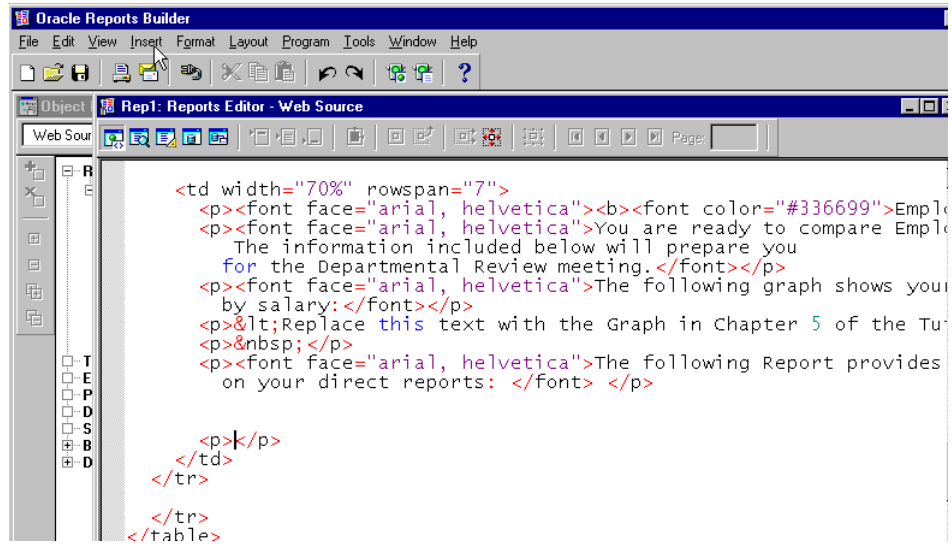
5. On the first page of the wizard, title your report "My Team's Salaries" and select the **Group Above** radio button, then click **Next**.

6. Let's format the data so that the employee names list in a downward column.

    On the Groups page, click **G_EMPLOYEE_ID,** then click **Down** to move the group to the Displayed Groups list.

    Note: If you did not select the fields in the correct order in Section 2.2, "Use the Data Wizard to add data to a sample Web page", you may not see this group. If this is the case, copy the SQL text from tutorial_sql.txt into the Query Statement box.

    ---

7. Click **Next**.

8. On the Fields page, click the double arrows (**>>**) to move all **Available Fields** to the **Displayed Fields** list.

9. In the Displayed Fields list, click EMPLOYEE_ID1, then click the left arrow (**<**) to move the field back to the Available Fields list, then click **Next**.

10. On the Labels page, click **Next**.

11. Make sure the **Predefined Templates** radio button is selected on the Templates page.

12. Make sure the **Beige** template is selected, then click **Finish**.

The Report Wizard has now generated the HTML with custom JSP tags to display your formatted data on the Web page.

13. Choose **File > Save As**.

14. In the Save dialog box, change the name to `empreva_<your initials>.jsp` and click **Save**.

15. Click the **Run Web Layout** icon in the toolbar to run and view your report in your Web browser. Your report should look something like this:

> **Note:** If Netscape 7.0 is your default browser, the browser may not display. You can work around this bug by making a copy of the Netscape 7.0 executable, naming it netscape.exe; with this name, the browser will display as expected.

Figure 4–3   JSP-based Web Report with Report Block
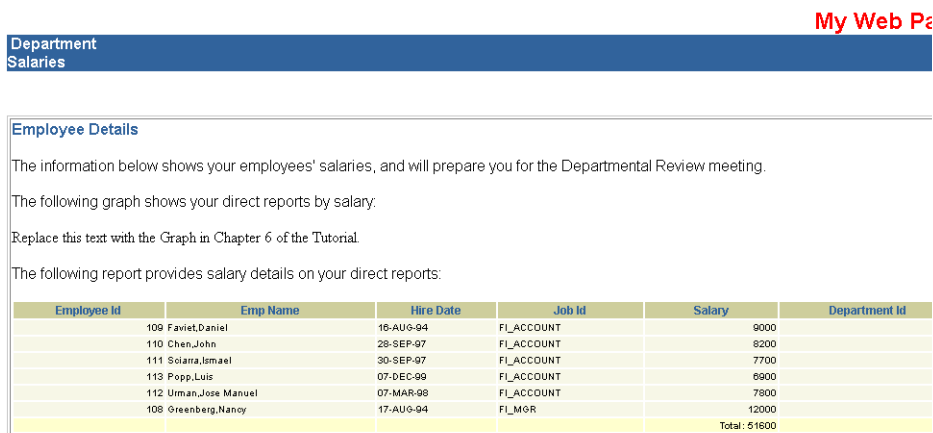
> **Note:** You can also run the `empreva.jsp` file we've provided by opening the JSP in Reports Builder, and running it to the Web.

4.2 Summary

Congratulations! You have added a report block to a Web report. You now know how to:

- Create a report block
- Add a report block to an HTML page for a JSP-based Web report

To review your results, continue on to Chapter 5, "Reviewing the Source Code for the Report Block". Otherwise, skip to Chapter 6, "Creating a Graph for the Web Report" to add a graph to your report.

5

Reviewing the Source Code for the Report Block

Estimated completion time: 5 minutes

In Chapter 4, "Creating a Report Block for the Web Report", you added a report block to a JSP-based Web report. This chapter reviews the source code added for the report block.

5.1 View the Web source in Reports Builder

To view the Web source:

- Open the Web source for the report you created in Chapter 4, "Creating a Report Block for the Web Report" called empreva_<your initials>.jsp by double-clicking the Web Source node in the Object Navigator.

5.1.1 Review the Header tag and body

1. In the Web Source view, locate the `<!-- Header -->` line.

   This HTML code indicates a comment line that helps you locate the report heading information in the file.

2. Locate the `<th>` and `<tr>` tags, which should look something like this:

```
<tr>
<th <rw:id id="HBEMPLOYEEID92" asArray="no" />
    class="OraColumnHeader">Employee Id </th>
<th <rw:id id="HBempname92" asArray="no" /> class="OraColumnHeader"> Emp
    Name </th>
...
```

5.1.2 Review the rw:foreach tag and body

The `rw:foreach` tag loops across a group. The layout is repeated for each instance of the specified group.

1. In the Web Source view, locate the `<rw:foreach>` JSP tag and find this line:

    ```
    <rw:foreach id="R_G_EMPLOYEE_ID921" src="G_EMPLOYEE_ID">
    ```

    Here, `R_G_EMPLOYEEID` is a unique identifier for this loop and `G_EMPLOYEE_ID` is the data model group on which the loop repeats.

2. Notice that the `</tbody>` tag that closes the table follows the closing `</rw:foreach>` tag, shown here:

    ```
        <td <rw:headers id=HFDEPARTMENTID92" src...>
      </tr>
      </rw:foreach>
    </tbody>
    ```

5.2 Verify your JSP code

If you've gone through this chapter and still aren't sure whether your JSP is correct, open the file we've provided, called `empreva.jsp`. This file is located in your examples directory.

5.3 Summary

Congratulations! You have finished reviewing the source code for the report block you added in Chapter 4, "Creating a Report Block for the Web Report". Continue to Chapter 6, "Creating a Graph for the Web Report" to add a graph to your Web report.

6

Creating a Graph for the Web Report

Estimated completion time: 15 minutes

This chapter describes how to use the Graph Wizard to create a graph that will enable managers to view their employees' salaries, and where each employee's compensation rate falls within a specific department.

If you already know how to use the Graph Wizard, refer to Appendix A.3, "Entries for the Graph Wizard"for a quick reference guide.

Figure 6–1   Adding a Graph to a JSP-based Web Report

6.1 Open the source for the report in Reports Builder

To open the Web source in Reports Builder:

1. In the Object Navigator, make sure the file `empreva_<your initials>.jsp` is open.

> **Note:** If you have not already created your own JSP-based Web report, open the file we've provided, named `empreva.jsp`.

2. In the Object Navigator, under the report name, double-click the Web Source icon to display the source code in the Web Source view.

3. In the Web Source view, click **Edit > Find and Replace** to find and delete the following text:

```
Replace this text with the Graph in Chapter 6 of the Tutorial.
```

6.2 Create a graph using the Graph Wizard

In this section, you will use the Graph Wizard to insert the JSP and XML code for the graph into the Web page for your Web report. The steps show you how to choose the type of graph you want, and the definitions of the X and Y axes. Note that you will base the graph on the same query you created in Section 2.2, "Use the Data Wizard to add data to a sample Web page".

If you already know how to use the Graph Wizard, you can refer to Appendix A, "Quick Reference Guide" for the entries for the Graph Wizard.

To create a graph:

1. Make sure your cursor is where you deleted the text, and choose **Insert > Graph** to display the Graph Wizard.

Figure 6–2   Inserting a Graph into your Web Source
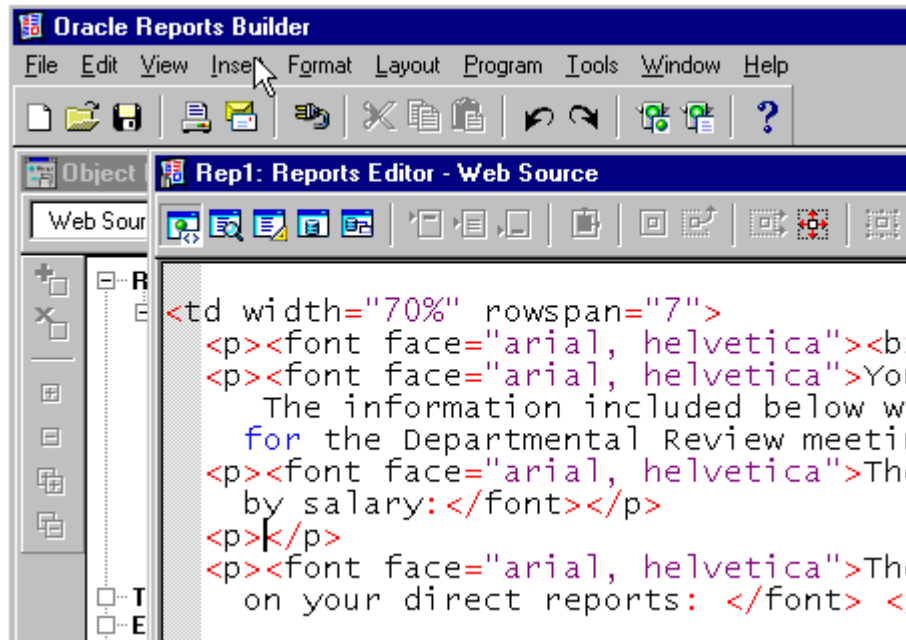
2. If the Welcome page displays, click **Next**.

3. On the next page of the Graph Wizard, notice that you can choose from a variety of graph styles. For our simple report, let's choose a Bar graph.

    In the Graph Wizard, make sure the default graph type (Bar) is selected, then click **Next**.

4. Move **EMPLOYEE_ID** to the **X-Axis Categories** list, then click **Next**.

5. Move **SALARY** to the **Y-Axis Data** list, then click **Next**.

6. On the **Layout** page, click and drag the **EMPLOYEE_ID** field from the Groups field to the Bars field, then click **Next**.

7. On the **Graph Titles** page, select **Show Title**, and in the first text box, type `Employees by Salary`, then click **Next**.

8. Make sure **Show Legend** is selected.

9. From the **Location** list, choose **Right**, then click **Next**.

10. In the **Show X-Axis title** field, type `Employees`, then click **Next**.

Note: **Show X-Axis Title** is automatically selected when you type text into the field.

---

11. In the **Show Y1-Axis title** field, type `Salaries`, then click **Next**.

12. At the bottom of the page in the Graph Wizard, click the **Row 1 Color** and choose a different color, for example dark pink.

Choose other colors for the other rows, if desired.

13. Click **Finish**.

Note: If you click **Next** instead of **Finish**, you'll notice there's another page in the Graph Wizard where you can add hyperlinks to your graph. Since we are not using graph hyperlinks in this tutorial, you can simply click **Finish**. If you'd like to learn more about using the Graph Wizard and graph hyperlinks, visit *Getting Started with Oracle Reports* and navigate to the **Examples** page. Here, you can view an example on using graph hyperlinks.

---

14. Click the **Run Web Layout** icon in the toolbar to preview the report in your Web browser.

The report should look something like this:

Note: If Netscape 7.0 is your default browser, the browser may not display. You can work around this bug by making a copy of the Netscape 7.0 executable, naming it netscape.exe; with this name, the browser will display as expected.

Figure 6–3   JSP-based Web Report with Graph

6.3 Save your report with the new graph

Saving your report with a different filename enables you to compare your results with those we've provided with the tutorial. This way, you can also roll back to the results of a previous chapter, if necessary.

1. Choose **File > Save As**.

2. In the Save dialog box, change the name to `emprevb_<your initials>.jsp` and click **Save**.

6.4 Summary

Congratulations! You have added a graph to a JSP-based Web report. You now know how to:

■ Use the Graph Wizard to design and add a graph to a Web report

To review your results, continue to Chapter 7, "Reviewing the Source Code for the Graph".

For information on adding a JSP parameter form to this Web report, refer to the *Oracle Reports Building Reports* manual. For information on deploying this report to the Web, refer to the *Oracle Application Server Reports Services Publishing Reports to the Web* manual.

7

Reviewing the Source Code for the Graph

Estimated completion time: 5 minutes

In Chapter 6, "Creating a Graph for the Web Report", you added a graph to a JSP-based Web report. This chapter reviews the source code added for the graph.

7.1 View the source in Reports Builder

Open the Web Source view for the report you created in Chapter 6, "Creating a Graph for the Web Report" called emprevb_<your initials>.jsp.

7.2 Review the rw:graph Tag

The rw:graph tag brackets the graph information and links the graph to the data source. It also identifies the categories and the data fields.

1. In the Web Source view, locate the <rw:graph> JSP tag.

   The code within the rw:graph tag is XML.

   ```
   <rw:graph id="graph" src="G_EMPLOYEE_ID" series="EMPLOYEE_ID"
   dataValues="SALARY">
   ```

   The series tag defines the source for the values along the X-axis, and the dataValues tag defines the source for the data along the Y-axis. You defined these parameters in the Graph Wizard in Chapter 6.

2. Locate the <SeriesItems> tag below the rw:graph tag:

   ```
   <SeriesItems>
   <Series id="0" color="#cc66cc"/>
   </SeriesItems>
   ```

This tag represents the modification to the Row 1 color we made in the Graph Wizard in Chapter 6. If you chose a different color, you will see a different value for the color tag.

3. Here, you can see that the color value #cc66cc is applied to the first bar along the x-axis:

Figure 7–1   Graph with Color on first row

4. Locate the `<Title>` tag, located here:

```
</SeriesItems>
<Title visible="true" text="Employees by Salary"/>
```

This tag adds the graph title to your Web report, shown here:

Figure 7–2 Graph with Title
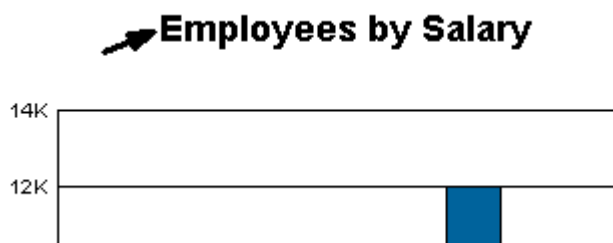
5. Locate the `</rw:graph>` JSP tag, located here:

```
</Graph>

-->
</rw:graph></p>
  <p></p>
...
```

The XML that produces the graph is closed before the `</rw:graph>` JSP tag is closed.

> **Note:** The Graph Wizard is re-entrant. So, if you'd like to modify your graph, you can always move your cursor into the XML between the rw:graph tags, then choose **Edit > Selection**. The Graph Wizard displays with the options you chose in Chapter 6.

7.3 Summary

Congratulations! You have finished reviewing the source code for your new graph. For more information on creating Web reports, adding report blocks and data, and creating graphs, see the *Reports Builder online help*.

8

Generating a Paper Report

Estimated Completion Time: 15 minutes

This chapter describes how to create a paper version of the Web report you just created. Here, you will use the Report Wizard to create a paper layout using the same data model you created in Chapter 2, "Adding Data to a Report".

If you already know how to use the Report Wizard, refer to Appendix A.4, "Entries for the Report Wizard"for a quick reference guide.

Figure 8–1   Generating a Paper Report from a JSP-based Web Report Data Model
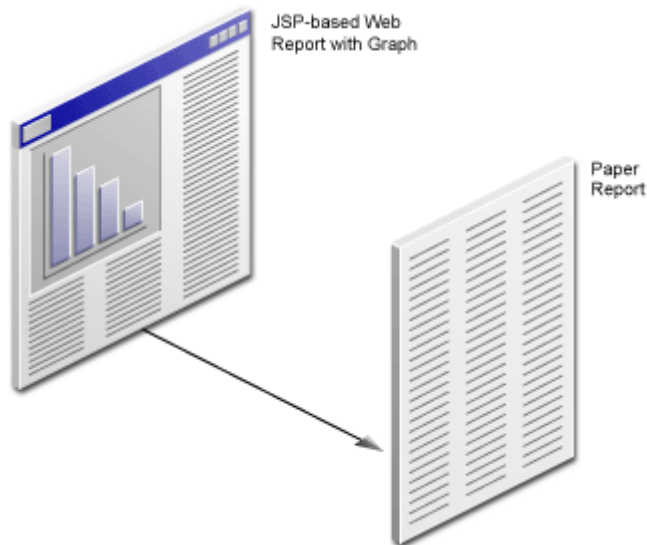
8.1 Generate a paper report based on your data model

In this section, you will learn how to generate a paper layout based on the same data model you created in Chapter 2, "Adding Data to a Report".

Note: You must be connected to the same database you used, and access the Human Resources (HR) schema. If you do not know the connection information for the database, contact your database administrator.

To generate a paper report: using the Report Wizard:

1. In the Object Navigator, make sure the report you created in Chapter 6, "Creating a Graph for the Web Report" called emprevb_<your initials>.jsp is open.

Note: If you don't want to create a JSP-based Web report first, and just want to learn how to generate a paper layout for a JSP-based Web report, open the file we've provided, called emprevb.jsp.

---

2. In the Object Navigator, right-click the report name.

3. From the pop-up menu, choose **Report Wizard**.

4. Let's choose the Paper Layout since we're generating a paper report. We don't need to select a Web Layout since we've already done that in Chapter 3.

    In the Report Wizard, on the Report Type page, select the **Create Paper Layout only** radio button.

5. On the Style page, make sure the **Group Above** radio button is selected, and that the title is "My Team's Salaries."

6. Click the **Data** tab.

    The data model you created in Chapter 2, "Adding Data to a Report" should display in the SQL Query Statement text box.

7. On the Fields page, move all the fields back to the **Available Fields** list, then move the following fields to the **Displayed Fields** list by selecting the field, then clicking **>**.

    ■ emp_name

- mgr_name

- SALARY

- SumSALARYPermgr_name

8. On the Labels page, in the **Label** field for the emp_name field, type `Employee Name`.

9. In the Label field for the mgr_name field, type `Manager Name`.

10. In the Label field for the SumSALARYPermgr_name field, type `Total Salary`.

11. On the Templates page, make sure the **Beige** template is selected.

12. Click **Finish**.

The report displays in the Paper Design view, and should look something like this:

Figure 8–2   Final Paper Report

13. Choose **File > Save As**.

14. Change the name to `emprev_paper_<your initials>.rdf`, and make sure "rdf" is selected.

15. Click **Save**.

8.2 Modifying a report in the Paper Design view

Reports Builder provides you with numerous tools you can use in the Paper Design view to modify the look and feel of your report. Because you are editing live data, you can see exactly how the end result of your report will appear. This section describes how to enhance your report using some of the most commonly used formats:

- aligning columns

- setting format masks

- manipulating objects

- editing text

- modifying visual attributes

- highlighting data

- inserting page numbering

- inserting current data and time

To modify the appearance of your report in the Paper Design view:

1. In the Paper Design view, click the Flex Off button in the toolbar.

2. Align the title with the logo.

Click the report title "My Team's Salaries" and drag it 1.5 inches to the left, and 0.25 inches down.

3. While the object is selected, choose **Format > Font**, then choose Arial, Bold, 12pt to format the text.

> **Tip:** If the text no longer fits within the object area, click the text object again, and drag one of the black squares to the right.

4. Your report title should now look like this:

Figure 8–3   Formatted title of the paper report

5. Click the number column beneath the Salary label. Notice how all the number values are selected.

6. In the toolbar, click the Currency button once, click the Commas button once, then click the Add Decimal Place button twice.

7. While the number values are still selected, click the Align Right button in the toolbar.

8. The Salary column should now look like this:

Figure 8–4   Formatted numbers in the Salary column

9. Since the salary numbers are aligned to the right, now align the Salary label to the right.

    To do so, click the Salary label, then click the Align Right button in the toolbar.

10. Now, format the Total Salary numbers in the same way the Salary numbers are formatted.

11. Add space between the Manager label and the manager's name. To do so, click a manager's name (e.g., Greenberg,Nancy). While the object is selected, use the right arrow on your keyboard to move the field to the right.

Your report should now look something like this:

Figure 8–5   Final formatted paper report

8.3 Summary

Congratulations! You have now generated a paper report based on the data model you created for a JSP-based Web report. You now know how to:

- Open the Report Wizard for an existing JSP-based Web report

- Create a paper layout for your report

- Produce a paper version of your Web report

- Format a paper report to make it more readable

For more information on generating a paper report based on an existing data model, see the *Reports Builder online help*.

A

Quick Reference Guide

This appendix contains tables that describe the entries you make into each wizard in Reports Builder. These tables are meant to be a quick reference guide for you to use later, or for you to use if you're already familiar with the wizards in Reports Builder.

A.1 Entries for the Data Wizard

The following table shows the entries you made in the Data Wizard in Chapter 2, "Adding Data to a Report".

Table A–1   Data Wizard Input

Data Wizard Page	Field	Input
Query	Name	Employee Salaries
Data Source	SQL Query	
Data	Query Statement	Paste in SQL from:
		<example files>/tutorial_sql.txt.
	Connect	Obtain the connection string from your DBA
Group	Group Fields	MGR_NAME
Totals	Total Fields	Sum(SALARY)

A.2 Entries for the report block

The following table shows the entries for the report block in Chapter 4, "Creating a Report Block for the Web Report".

Table A–2   Report Wizard Input for the Report Block

Report Block Wizard Page	Field	Input
Title	Title	Direct Reports by Manager
Type	Group Above	Select
Groups	Available Groups	G_EMPLOYEE_ID, DOWN
Displayed Fields	Available Fields	All fields (move all fields to Displayed)
Totals		None
Templates	Predefined Templates	Beige

A.3 Entries for the Graph Wizard

The following table shows the entries you made to the Graph Wizard in Chapter 6, "Creating a Graph for the Web Report".

Table A–3   Graph Wizard Input

Graph Wizard Page	Field	Input
Type	Type	Bar
X-Axis -Category	Available Columns	Move EMPLOYEE_ID and MGR_NAME to X-Axis Categories
Y-Axis - Data	Available Columns	Move SALARY to Y-Axis Data
Layout	Groups field	Drag EMPLOYEE_ID from the Groups field to the Bars field.
Title	Show Title	Check the box and type Salary Comparison
Legend	Show Legend	Select
X-Axis	Show X-Axis Title	Check the box and type Employees
Y-Axis	Show Y-Axis Title	Check the box and type Salaries
Plot Area Options	Row 1 Color	Select a different color

A.4 Entries for the Report Wizard

The following table shows the entries for the Report Wizard in Chapter 8, "Generating a Paper Report".

Table A–4   Report Wizard Input for the Report Block

Report Wizard Page	Field	Input
Layout	Create Paper Layout Only	Select
Type	Group Above	Select
Groups	Available Groups	(Do not change)
Displayed Fields	Available Fields	emp_name
		mgr_name
		SumSALARYPermgr_name
		SALARY
Totals		(Do not change)
Labels	All labels	Change labels to be meaningful.
Templates	Predefined Templates	Beige

B

Tool Palette and Toolbar Reference

This appendix provides descriptions of the buttons and tools in the Reports Builder tool palettes and toolbars.

B.1 Main Toolbar

The main toolbar is located at the top of the Reports Builder window, directly beneath the menu bar:

Figure B–1   Main Toolbar
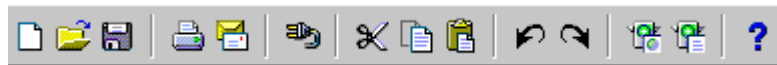
New button. Displays the New Report dialog box.

Open button. Displays the Open dialog box.

Save button. Saves the report. If you haven't saved the report before, the Save As dialog box displays.

Print button. Prints the paper report.

Mail button. Displays the Mail dialog box.

Connect button. Displays the Connect dialog box.

Cut button. Deletes the currently selected item and temporarily places it in the clipboard. Use Paste to paste the selected item.

Copy button. Temporarily places a copy of the selected item in the clipboard. Click the **Paste** button to paste the selected item.

Paste button. Pastes the item in the clipboard in current location of the cursor.

Undo button. Undoes the last action performed.

Redo button. Performs the last action again.

Run Web Layout button. Runs the current report to your Web browser.

Run Paper Layout button. Runs the current report to the Paper Design view in Reports Builder.

Help button. Displays the *Reports Builder online help*.

B.2 Data Model view tool palette

The Data Model view tool palette is a vertical group of tools located on the left-hand side of the Data Model view.

Select tool. Deselects any selected tool to "turn off" the current tool.

Magnify tool. Zooms in the view on the clicked object. Use **SHIFT + Magnify** to zoom out.

Summary Column tool. Creates a summary column in the query.

Data Link tool. Creates a link between the columns in the queries.

Formula Column tool. Creates a formula column in the query.

Cross Product tool. Creates a matrix (cross-product) group.

Placeholder tool. Creates a placeholder column which you can modify later.

SQL Query tool. Displays the SQL Query Statement dialog box where you can enter a SQL query SELECT statement or use Query Builder to create a query.

Ref Cursor tool. Displays the PL/SQL Editor where you can type a ref cursor query.

XML Query tool. Displays the Define XML Query dialog box, where you can specify the XML data definition and data source.

JDBC Query tool. Displays the JDBC Query dialog box, where you can define the SQL or stored procedure to define the data for the query.

Text Query tool. Displays the Text Query dialog box, where you can specify a text data definition and data source.

Express Server Query tool. Displays the Express Server Query dialog box, where you can specify an Oracle Express data definition and data source.

B.3 Paper Layout view tool palette

The Paper Layout view tool palette is a vertical group of tools located on the left-hand side of the Paper Layout view.

Select tool. Deselects any selected tool to "turn off" the current tool.

Magnify tool. Zooms in the view on the clicked object. Use **SHIFT + Magnify** to zoom out.

Frame Select tool. Selects all objects within the selected frame or repeating frame, depending upon their explicit anchors (first click the tool, then the frame).

Reshape tool. Enables you to reshape the selected boilerplate object.

A

Text tool. Creates a boilerplate text object.

Rotate tool. Enables you to rotate the direction of the selected boilerplate object.

Line tool. Draws a line boilerplate object.

Rectangle tool. Draws a rectangle boilerplate object.

Arc tool. Draws an arc boilerplate image.

Rounded Rectangle tool. Draws a rounded rectangle boilerplate object.

Polyline tool. Draws an open multi-lined boilerplate object. Use your mouse to create the multiple lines.

Polygon tool. Draws a multi-sided boilerplate object. The object must be closed, unlike a polyline object.

Freehand tool. Draws a line where you drag your mouse.

Ellipse tool. Draws an ellipse boilerplate object.

Frame tool. Draws a frame.

Repeating Frame tool. Draws a repeating frame.

Graph tool. Displays the Graph Wizard so that you can to define a graph that will be inserted into your layout.

Field tool. Creates a field object.

Anchor tool. Creates an anchor between two objects in your layout.

File Link tool. Creates a link file object that you can use to link an external file to your report.

Report Block tool. Displays the Report Block wizard so that you can add a new report block to your layout.

Glossary

column

1. A vertical space in a database table that represents a particular domain of data. A column has a column name (e.g., ENAME) and a specific datatype (e.g., CHAR). For example, in a table of employee information, all of the employees' names would constitute one column. A record group column represents a database column.

2. A data model object created automatically for each column expression in a query's SELECT list, or created manually to perform summaries, formulas, or act as a placeholder.

3. The representation of an attribute of an entity.

data model

A relational model that defines what data should be fetched from the data source(s), what values should be computed, and how data should be ordered in a report. Reports Builder objects that define the data model are queries, groups, columns, parameters, and links.

Data Model view

One of the views of the Report Editor that displays a structural representation of the data in a report. The objects do not appear in the report output, but the structure determines the layout style, and the data objects provide the values that appear in the layout objects.

database

1. A set of dictionary tables and user tables that are treated as a unit.

2. (Oracle Express) A single file (possibly accompanied by extension files) that contains objects that organize, store, and manipulate data. In Express, examples of such objects are variables, dimensions, formulas, models, and programs.

data source

A source for data returned by a query, including database objects such as tables, views, synonyms, snapshots, and queries stored as views. OracleAS Reports Services allows you to access any data source.

The new pluggable data source (PDS) architecture replaces Oracle Open Client Adapter (OCA), and the Open Database Connectivity (ODBC) drivers are no longer supported in Oracle Reports 10*g*. However, Java Database Connectivity (JDBC) is one of the pluggable data sources available that can utilize the JDBC-ODBC bridge, allowing access to other data sources.

detail query

When defining a master/detail report, the detail query retrieves all related records for each record retrieved by the master, or parent, query.

dialog box

A partial screen or window that prompts you to enter information necessary to complete an operation.

disabled

An interface element state that means a menu item, button, and so on, cannot be used in the current context (i.e., it does not respond to keyboard or mouse input).

editor

See view.

enabled

An interface element state that means that a menu item, button, and so on, can be used in the current context (that is, it responds to keyboard or cursor/mouse input).

field

1. An interface element in which you enter, edit, or delete data.

2. A layout object that defines how the data for a specific query column appears.

foreign key

A value or column in one table that refers to a primary key in another table.

format mask

A setting that defines the appearance of the value of a field. For example, a format mask is used to specify the display of currency amounts and dates.

format trigger

A PL/SQL function that allows you to dynamically change the formatting attributes of an object.

formula column

A user-created column that gets its data from a PL/SQL function or expression, a SQL statement, or a combination of these.

frame

A layout object used to enclose other layout objects and control the formatting, frequency, and positioning of several objects simultaneously.

group

1. In Reports Builder, a data model object that is created automatically to contain all the columns selected by a query, or created by the user to modify the hierarchy of the data appearing in a report; it is used primarily for creating breaks in a report, as well as for resetting computations.

2. An object that is composed of several other objects.

HTML (Hypertext Markup Language)

Acronym for Hypertext Markup Language. A tag-based ASCII language used to specify the content and links to other documents on Web servers on the Internet. End users with Web browsers view HTML documents and follow links to display other documents.

hyperlink

A reference (link) from some point in one document to (some point in) another document or another place in the same document. A Web browser usually displays a hyperlink in some distinguishing way (in a different color, font or style). When users activate hyperlinks (by clicking on them with a mouse) the browser displays the target of the link.

icon

A graphic representation of a window or tool.

image

A bitmapped object that can be stored and loaded into an application. The client cannot modify an imported image.

intranet

An internal TCP/IP network, access to which is restricted (via a firewall) to individuals inside the company or organization. An intranet provides similar services within an organization to those provided by the Internet, but is not necessarily connected to the Internet. A common example of an intranet is when a company sets up one or more Web servers on an internal network for distribution of information or applications within the company.

Java

A computer language that supports programming for the Internet in the form of platform-independent "applets".

JSP (JavaServer Page)

JavaServer Page (JSP) technology is an extension to the Java Servlet technology from Sun Microsystems that provides a simple programming vehicle for displaying dynamic content on a Web page. JSP is a server-side technology. A JSP is an HTML page with embedded Java source code that is executed in the Web server or application server. The HTML provides the page layout that is returned to the Web browser, and the Java provides the business logic.

layout

See Paper Layout view.

margin

An optional report region that appears at the top and bottom of each logical page in a report section (Header, Main, or Trailer). The margin may include any layout object, but typically contains boilerplate and fields (for page numbers, page totals, grand totals, and current date and time).

object

1. An item that can be placed on the layout. The following are examples of objects: rectangle, line, ellipse, arc, polygon, polyline, rounded rectangle, freehand, chart, text, symbol, and text field.

2. In an Oracle database, an instance of an object type. An object can be a row in an object table, or the portion of a row contained in a column object in a relational table.

Object Navigator

A hierarchical browsing and editing interface that enables you to locate and manipulate application objects quickly and easily. Features include:

- A hierarchy represented by indentation and expandable nodes (top-level nodes show module types, database objects, and built-in packages), enabling tasks such as creating, editing, renaming, and deleting objects.

- A find field and icons, enabling forward and backward searches for any level of node or for an individual item in a node

- Icons in the horizontal toolbar replicating common File menu functions

Oracle Application Server (OracleAS)

A strategic platform for network application deployment. By moving application logic to application servers and deploying network clients, organizations can realize substantial savings through reduced complexity, better manageability, and simplified development and deployment. OracleAS provides the only business-critical platform that offers easy database Web publishing and complete legacy integration while transitioning from traditional client-server to network application architectures.

Oracle Developer Suite

Combines leading Oracle application development and business intelligence tools into a single, integrated product. Built on Internet standards such as Java and XML, the suite provides a complete and highly productive development environment for building applications for Oracle Application Server and the Oracle database.

ORACLE_HOME

An alternate name for the top directory in the Oracle directory hierarchy on some directory-based operating systems. An environment variable that indicates the root directory of Oracle products.

You can refer to the directory specified by *ORACLE_HOME* in syntax:

On UNIX: $ORACLE_HOME

On Windows: %ORACLE_HOME%

OracleAS Portal

An HTML-based development tool for building scalable, secure, extensible HTML applications and Web sites. OracleAS Reports Services uses OracleAS Portal to control end user access to reports published on the Web by storing information about report requests, the secured server, and any OracleAS Reports Services printer used to print report output.

OracleAS Reports Services

See Reports Services.

Paper Design view

One of the views of the Report Editor that displays output for paper reports and allows you to make many commonly required, simple modifications to the layout, such as spacing, formatting fields, color, and editing text, without having to open the Paper Layout view.

Paper Layout view

One of the views of the Report Editor that displays the layout objects in a paper report and allows you to make many modifications to any layout object. All layout objects have properties that you can modify using the Property Inspector. The hierarchy of the layout objects is determined by the Data Model.

Paper Parameter Form view

Displays the layout of the Parameter Form that, at runtime, allows user input of parameter values in the Runtime Parameter Form.

PDF (Portable Document Format)

Acronym for Portable Document Format. A file format (native for Adobe Acrobat) for representing documents in a manner that is independent of the original application software, hardware, and operating system used to create the documents. A PDF file can describe documents containing any combination of text, graphics, and images in a device-independent and resolution independent format.

PL/SQL

Oracle's proprietary extension to the SQL language. Adds procedural and other constructs to SQL that make it suitable for writing applications.

Property Inspector

A window that enables you to view, locate, and set the properties of the currently selected object(s) in the Object Navigator, Report Editor, and Template Editor.

Every Reports Builder object (query, group, frame, parameter, etc.) has associated properties that can be viewed using the Property Inspector. The Property Inspector features:

- expandable and collapsible nodes

- in-place property editing

- search features

- multi-selection

- complex property dialogs

- the ability to invoke multiple instances of the Property Inspector

To get help on any property, click the property in the Property Inspector and press F1.

query

A SQL SELECT statement that specifies the data you wish to retrieve from one or more tables or views of a database.

RDF file

A file that contains a single report definition in binary format. .RDF files are used to both run and edit reports.

record

One row fetched by a SQL SELECT statement.

REP file

A file that contains a single report definition in binary format. .REP files are used solely to run reports; you cannot edit a .REP file.

repeating frame

A layout object used to display rows of data that are fetched for a group.

Report Editor

The Reports Builder window that provides different views to help you handle the data objects and layout objects for Web and paper reports. The views are:

- Data Model view

- Paper Layout view

- Paper Design view
- Paper Parameter Form view
- Web source view

Reports Builder (rwbuilder)

An Oracle Reports executable that starts Reports Builder to enable report developers to create and maintain report definitions.

Reports Runtime (rwrun)

An Oracle Reports executable that runs a report using the OracleAS Reports Services in-process server.

Reports Services

The runtime environment for Reports Developer applications. OracleAS Reports Services executes, distributes, and publishes your reports for enterprise wide reporting. Using OracleAS Reports Services to deploy your reports results in gains of flexibility, time savings, and processing capacity.

Reports Servlet (rwservlet)

An Oracle Reports executable that translates and delivers information between HTTP and the Reports Server, enabling you to run a report dynamically from your Web browser.

row

One set of field values in a table; for example, the fields representing one employee in the example table EMP.

Runtime Parameter Form

A screen or window appearing optionally at runtime in which a user can modify print options and parameters prior to report execution.

schema

A collection of related database objects, usually grouped by database user ID. Schema objects include tables, views, sequences, stored program units, synonyms, indexes, clusters, and database links.

SELECT statement

A SQL statement that specifies which rows and columns to fetch from one or more tables or views.

SQL

A standard interface for storing and retrieving information in a relational database. SQL is an acronym for Structured Query Language.

SQL file

A file that contains a query stored in text (e.g., ASCII or EBCDIC) format.

SQL script

A file containing SQL statements that you can run to perform database administration quickly and easily. Several SQL scripts are shipped with Oracle products.

SQL statement

A SQL instruction to Oracle. A SELECT statement is one type of SQL statement.

style sheet

HTML extensions that provide powerful formatting flexibility in HTML documents. To view an HTML document that takes advantage of style sheets, display it in a browser that supports style sheets.

table

A named collection of related information, stored in a relational database or server, in a two-dimensional grid that is made up of rows and columns.

tabular

A default layout displaying labels at the top of the page and rows of data underneath the labels.

template

A skeleton definition containing common style and standards, and may include graphics. A template provides a standard format to enable quick and easy development of professional standard look-and-feel reports.

Template Editor

A work area in which you can define objects and formatting properties for your templates. It is similar to the Paper Layout view of the Report Editor. You can create, delete, and modify objects (e.g., page numbers, text, and graphics) in the margin area. You cannot create and delete objects in the body area, but you can modify the properties of body objects in the Property Inspector.

tool

An iconic button used to create and manipulate objects in an application.

tool palette

A collection of tools represented by iconic buttons in the user interface that allow a report developer to perform tasks, such as drawing a rectangle in the Paper Layout view or creating a query in the Data Model view.

toolbar

A collection of iconic buttons that perform product commands. Usually aligned horizontally along the top, or vertically down the side of a window.

URL (Uniform Resource Locator)

A compact string representation of the location for a resource that is available through the Internet. It is also the text string format clients use to encode requests to OracleAS.

view

1. In Reports Builder, a work area in which you perform a specific set of tasks, such as defining a report data model, layout, or Parameter Form.

2. 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.

Web browser

A program that end users utilize to read HTML documents and programs stored on a computer (serviced by a Web server).

Web server

A server process (HTTP daemon) running at a Web site which sends out Web pages in response to HTTP requests from remote Web browsers.

Web source view

One of the views of the Report Editor that displays the HTML / JSP source for a report. You can use this view to add dynamic content to a Web page using the Report Block Wizard and the Graph Wizard. Experienced Java developers can edit the Web source directly in this view.

wizard

A step-by-step interface for commonly performed tasks. The wizards in Reports Builder are:

- Report Wizard: guides you through the steps to create a basic paper or Web report. Each page of the wizard asks you for information to help you create your initial report.

- Data Wizard: helps you helps you quickly define or modify a query for a multiquery data models.

- Graph Wizard: Adds variety of charts and graphs, including true 3-dimensional graphs. Implemented in Reports Builder with the Oracle BI graph bean.

- Report Block Wizard: enables you to add data to a static HTML page.

XML

Acronym for Extensible Markup Language. A metalanguage using SGML to define and structure data. Reports Builder supports XML output to enable Web publishing as well as electronic data exchange with third-party applications. You can also use XML to build report definitions that can be merged with other report definitions at runtime or run separately.

Index

S

saving,   6-6
   JSP-based Web report
      saving,   6-6
scenario,   1-1
seriesitem tag
   reviewing,   7-1

T

table designs
   FormalKey,   A-2, A-3

W

Web page
   opening in Reports Builder,   2-2
Web source
   reviewing,   3-1
   reviewing a graph,   7-1
   reviewing a report block,   5-1