11.5 Example: JSP Expressions

Listing 11.3 gives an example JSP page called `Expressions.jsp`. We placed the file in a directory called `jsp-scripting`, copied the entire directory from our development directory to the top level of the default Web application (in general, to the top-level directory of the Web application—one level up from `WEB-INF`), and used a URL of `http://host/jsp-scripting/Expressions.jsp`. Figures 11-2 and 11-3 show some typical results.

Figure 11-2. Result of `Expressions.jsp` using Macromedia JRun and omitting the `testParam` request parameter.

Figure 11-3. Result of `Expressions.jsp` using Caucho Resin and specifying `testing` as the value of the `testParam` request parameter.

Notice that we include `META` tags and a style sheet link in the `HEAD` section of the JSP page. It is good practice to include these elements, but there are two reasons why they are often omitted from pages generated by normal servlets.

First, with servlets, it is tedious to generate the required `println` statements. With JSP, however, the format is simpler and you can make use of the code reuse options in your usual HTML building tools. This convenience is an important factor in the use of JSP. JSP pages are not more *powerful* than servlets (they *are* servlets behind the scenes), but they are sometimes more *convenient* than servlets.

Second, servlets cannot use the simplest form of relative URLs (ones that refer to files in the same directory as the current page), since the servlet directories are not mapped to URLs in the same manner as are URLs for normal Web pages. Moreover, servers are expressly prohibited

from making content in `WEB-INF/classes` (or anywhere in `WEB-INF`) directly accessible to clients. So, it is impossible to put style sheets in the same directory as servlet class files, even if you use the `web.xml servlet` and `servlet-mapping` elements (see Section 2.11, "Web Applications: A Preview") to customize servlet URLs. JSP pages, on the other hand, are installed in the normal Web page hierarchy on the server, and relative URLs are resolved properly as long as the JSP page is accessed directly by the client rather than indirectly by means of a `RequestDispatcher` (see Chapter 15, "Integrating Servlets and JSP: The Model View Controller (MVC) Architecture").

Thus, in most cases style sheets and JSP pages can be kept together in the same directory. The source code for the style sheet, like all code shown or referenced in the book, can be found at http://www.coreservlets.com.

Listing 11.3 Expressions.jsp

```html
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<HTML>
<HEAD>
<TITLE>JSP Expressions</TITLE>
<META NAME="keywords"
      CONTENT="JSP,expressions,JavaServer Pages,servlets">
<META NAME="description"
      CONTENT="A quick example of JSP expressions.">
<LINK REL=STYLESHEET
      HREF="JSP-Styles.css"
      TYPE="text/css">
</HEAD>
<BODY>
<H2>JSP Expressions</H2>
<UL>
  <LI>Current time: <%= new java.util.Date() %>
  <LI>Server: <%= application.getServerInfo() %>
  <LI>Session ID: <%= session.getId() %>
  <LI>The <CODE>testParam</CODE> form parameter:
      <%= request.getParameter("testParam") %>
</UL>
</BODY></HTML>
```