[Team LiB] PREVIOUS NEXT

10.1 The Need for JSP

"Hey!" you say, "You just spent several chapters extolling the virtues of servlets. I like servlets. Servlets are convenient to write and efficient to execute. They make it simple to read request parameters and to set up custom code to handle missing and malformed data. They can easily make use of HTTP request headers and can flexibly manipulate HTTP response data. They can customize their behavior based on cookies, track user-specific data with the session-tracking API, and talk to relational databases with JDBC. What more do I need?"

Well, this is a good point. Servlets are indeed useful, and JSP by no means makes them obsolete. However, look at the list of topics for which servlets excel. They are all tasks related to *programming* or data *processing*. But servlets are not so good at *presentation*. Servlets have the following deficiencies when it comes to generating the output:

- **It is hard to write and maintain the HTML.** Using `print` statements to generate HTML? Hardly convenient: you have to use parentheses and semicolons, have to insert backslashes in front of embedded double quotes, and have to use string concatenation to put the content together. Besides, it simply does not look like HTML, so it is harder to visualize. Compare this servlet style with Listing 10.1 where you hardly even notice that the page is not an ordinary HTML document.

- **You cannot use standard HTML tools.** All those great Web-site development tools you have are of little use when you are writing Java code.

- **The HTML is inaccessible to non-Java developers.** If the HTML is embedded within Java code, a Web development expert who does not know the Java programming language will have trouble reviewing and changing the HTML.

[Team LiB] PREVIOUS NEXT