

[\[Team LiB \]](#)
[◀ PREVIOUS](#) [NEXT ▶](#)

15.8 Including Pages

The `forward` method of `RequestDispatcher` relies on the destination JSP page to generate the *complete* output. The servlet is not permitted to generate any output of its own.

An alternative to `forward` is `include`. With `include`, the servlet can combine its output with that of one or more JSP pages. More commonly, the servlet still relies on JSP pages to produce the output, but the servlet invokes different JSP pages to create different *sections* of the page. Does this sound familiar? It should: the `include` method of `RequestDispatcher` is the code that the `jsp:include` action ([Section 13.1](#)) invokes behind the scenes.

This approach is most common when your servlets create portal sites that let users specify where on the page they want various pieces of content to be displayed. Here is a representative example.

```
String firstTable, secondTable, thirdTable;
if (someCondition) {
    firstTable = "/WEB-INF/Sports-Scores.jsp";
    secondTable = "/WEB-INF/Stock-Prices.jsp";
    thirdTable = "/WEB-INF/Weather.jsp";
} else if (...) { ... }
RequestDispatcher dispatcher =
    request.getRequestDispatcher("/WEB-INF/Header.jsp");
dispatcher.include(request, response);
dispatcher =
    request.getRequestDispatcher(firstTable);
dispatcher.include(request, response);
dispatcher =
    request.getRequestDispatcher(secondTable);
dispatcher.include(request, response);
dispatcher =
    request.getRequestDispatcher(thirdTable);
dispatcher.include(request, response);
dispatcher =
    request.getRequestDispatcher("/WEB-INF/Footer.jsp");
dispatcher.include(request, response);
```

[\[Team LiB \]](#)
[◀ PREVIOUS](#) [NEXT ▶](#)