c) **Cookie** never expire.
f) **HttpSessions** expire only when the browsing session ends or when the `invalidate` method is called.
g) The `HttpSession` method `getAttribute` returns the object associated with a particular name.

**ANSWERS TO SELF-REVIEW EXERCISES**

9.1 a) **Servlet** b) `doGet`, `doPost` c) `getWriter` d) `action` e) `localhost` f) `getName`, `setName` g) `HttpSession`

9.2 a) False. Servlets are usually used on the server side.
b) True.
c) False. The two most common HTTP request types are `get` and `post`.
d) False. The well-known port number for Web requests is 80.
e) False. **Cookies** expire when they reach their maximum age.
f) True.
g) True.

**EXERCISES**

9.3 Modify the **Cookie** example of Fig. 9.20 to list prices for each book in the book recommendations. Also, allow the user to select some or all of the recommended books and “order” them. Deploy your Web application on the Tomcat server.

9.4 Modify the **HttpSession** example of Fig. 9.24 to list prices for each book in the book recommendations. Also, allow the user to select some or all of the recommended books and “order” them. Deploy your Web application on the Tomcat server.

9.5 Create a Web application for dynamic FAQs. The application should obtain the information to create the dynamic FAQ Web page from a database that consists of a **Topics** table and an **FAQ** table. The **Topics** table should have two fields—a unique integer ID for each topic (`topicID`) and a name for each topic (`topicName`). The **FAQ** table should have three fields—the `topicID` (a foreign key), a string representing the question (`question`) and the answer to the question (`answer`). When the servlet is invoked, it should read the data from the database and return a dynamically created Web page containing each question and answer, sorted by topic.

9.6 Modify the Web application of Exercise 9.5 so that the initial request to the servlet returns a Web page of topics in the FAQ database. Then, the user can hyperlink to another servlet that returns only the frequently asked questions for a particular topic.

9.7 Modify the Web application of Fig. 9.27 to allow the user to see the survey results without responding to the survey.

9.8 Fig. 9.27 would allow users to vote as many times as they want by simply returning to the survey Web page and submitting additional votes. Modify your solution to Exercise 9.7 such that it uses cookies that last for one day to prevent users from voting more than once a day. When a user returns to the site, the cookie previously stored on their system is sent to the server. The servlet should check for the cookie and, if it exists, indicate that the client already voted in the last 24 hours. The servlet should also return the current survey results.

9.9 Modify the Web application of Fig. 9.27 to make it generic for use with any survey of the appropriate form. Use servlet parameters (as discussed in Section 9.8) to specify the survey options. When the user requests the survey, dynamically generate a form containing the survey options. Deploy this Web application twice using different context roots. **Note:** You may need to modify the database in this example so that it can store multiple surveys at once.