

# RQs + PEs: More Servlets

## Advanced Topics in Java

**Khalid Azim Mughal**

*khalid@ii.uib.no*

*<http://www.ii.uib.no/~khalid/atij/>*

*Version date: 2006-09-04*

## REVIEW QUESTIONS - More Servlets

1. Explain the term *redirection* in servlet response handling and how it works.
2. How can a servlet send a HTTP status code as a response?
3. Name and distinguish between the three *attribute scopes* in a web application, specially their creation and termination. Give examples of the kind of information one would store in these scopes.
4. Which methods can one use to handle the attributes in a scope?
5. What is a *session*? What is its purpose? Outline a typical session scenario.
6. How is a session obtained? And explain the functionality provided by the `HttpSession` interface.
7. Explain the states that a session goes through in its life cycle.
8. How are cookies used to support sessions?
9. Which scope would you use to share the following data:
  - the number of items that a client has accumulated in the shopping basket.
  - the connection to the database used in the application
  - A rebate code valid for a particular item, to be used only once when the item is selected.

10. Explain what is meant by *multi-thread servlet model*, with emphasis on thread-safety of data in the three scopes.
11. Explain *default URL*, *registered name* and *customized URL* when it comes to invoking a servlet.
12. Explain the purpose of *logging* and how this done by an application.
13. Explain how cookies are added to the response and read from the request. What must the servlet do if the information changes in a cookie that has been sent to the client previously?
14. Explain the procedure for specifying FORM-authentication for a secured servlet in the `web.xml` file that defines the servlet.
15. How are WAR files used for web application deployment?
16. Which of these statements are true. Explain your reasoning.
  - A servlet can only redirect to a passive resource.
  - A servlet can have several sessions at the same time.
  - An HTML form can only be retrieved from a file that has the HTML source code.
  - The `/servlet/` pattern can always be used to invoke a servlet.
  - The url pattern specified in the `web.xml` file can always be used to invoke the servlet associated with the pattern.
  - New checked exceptions can always be specified in the `throws` clause of an

overriding `doHttpRequestMethodName()` method.

- A cookie once sent to the client by a servlet is always returned by the client in all subsequent requests.
- All users who can access a secured servlet are defined in the `web.xml` file that defines the servlet.

## PROGRAMMING EXERCISES - More Servlets

1. Implement a shopping cart: from selection to checkout, with the possibility of amending the order. We ignore user authentication in this version.
  - A servlet presents an order form to the user. The servlet lists a finite number of items. For each item, it lists the quantity in stock and the price. Each item also has a Quantity required text field and a Add to Shopping Cart button. The servlet repeated allows the user to specify the quantity required for an item and add it to the shopping cart. When the Checkout button is clicked, the order is submitted.
  - A second servlet processes the order. It allows the user to go back and add more items, i.e a Continue shopping button. It also allows the user to change quantity and delete items. It only completes the order when the Confirm order button is clicked. After confirmation, the application resets to allow the processing of a new order.
  - In this version, the information about the items is read from a data file.
2. Extend the functionality of the shopping cart application in the previous exercise. Implement a servlet to present the information.
  - Track the number of times the application has been visited (i.e. no. of sessions that have been started at the site).
  - Track the number of orders that have been placed at the site (i.e. no. of sessions that resulted in confirmed orders).
  - Track the amount of money that has been spent on orders at the site.

- Track the top five items ordered at the site.
3. Extend the functionality of the shopping cart application in the previous exercises with user authentication. Users must register with password, and user information is stored in a data file, as is other pertinent information for the application. Implement the servlet(s) to present the information.
    - Track the number of times a user has visited the site.
    - Track the number of orders a user has placed.
    - Track the amount a user has spent on orders at the site.
    - Keep a Wish list of items for a user.
  4. Implement a web application for *dynamic* FAQs (*Frequently Asked Questions*).
    - The first servlet presents a dynamically generated list of topic categories.
    - The client can select a topic category to retrieve the FAQs for that particular category — generated by the second servlet.

The FAQ data is organized in two data files: `topics.data` and `faq.data`. A record in the `topics.data` file consists of two fields: a unique integer (`topicID`) which identifies a topic category (`topicCategory`). Each line specifies a single record, for example:

```
1001 HTTP Response Status Codes
1002 Main Servlet API Classes
1003 Reading a Request
```

## 1004 Reading a Response

A record in the `faq.data` file consists of three fields: a question (`question`), its answer (`answer`) and the topic ID (`topicID`) of it's category. Three lines specify a single record, for example:

```
What does the status code 400 signify?
```

```
The status code 400 (OK) signifies that the request succeeded normally.
```

```
1001
```

The application should read the FAQ data prior to handling client requests.

Create files with suitable FAQ data.

5. Extend the application in programming exercise 4 to use sessions and cookies to generate statistics:
  - Number of times a user has visited the FAQ site, which requires tracking sessions for the user.
  - Topics a user has visited previously, which requires tracking the topics visited by the user.
  - Top five topics, which requires tracking the topics visited by all users.
6. Implement a survey form, displaying the submitted information and other pertinent statistics.