Midterm review

Fall 2016

Devdatta Kulkarni

Midterm: Date, time, format

- Date and time:
  - Thursday, October 27, 5.30pm – 8.30pm
- Location: JGB 2.218 (Our classroom)
- Format:
  - Open book, open notes
  - Allowed:
    - Text books, class notes, example programs
    - e-book readers, if you have electronic version of the books and/or class notes, example programs in electronic format
  - Not allowed:
    - Querying search engines for answers
    - Running programs to find out answers

Midterm: Syllabus

- Everything that we have covered in class till this week
- Specific Topics:
  - Http, Servlets, Spring framework, dependency injection, unit testing, REST API design, RESTEasy framework, XML/HTML parsing, marshalling, unmarshalling, functional testing
- Book chapters:
  - Java for Web Applications: 1, 2, 3, 5, 12, 13, 14
  - RESTful Java with JAX-RS 2.0: 1, 2, 3, 4, 5, 6, 10

Revision

- Protocols/Specifications
  - HTTP
  - Servlets
- Software Architecture
  - Layered architecture
- Frameworks
  - Spring
  - RESTEasy
- Software Quality
  - Unit testing
  - Functional testing
- Tools
  - XML/HTML parsing

Protocols/Specifications

HTTP Protocol

- Main concepts
  - Request/response protocol
  - Versions 1.0, 1.1
  - Headers
    - Request/response headers
    - Cache control headers
  - Transfer-encoding: chunked
  - Session tracking mechanisms

HTTP Protocol

1. When is a particular HTTP header used?
2. What is the meaning of a particular HTTP header?
3. What will be response if the request contains a particular HTTP header?
4. As a web application developer what are some of the ways to ensure that the clients never have to work with stale data?
5. What mechanisms are available for tracking user session across requests?

Servlets

- Main concepts
  - Deployment descriptor
  - Servlet container
  - ServletContext
    - Represents the context for an entire web application (all Servlets)
    - Initialization information to *all* the servlets can be obtained from ServletContext
  - ServletConfig
    - Use to initialize a specific servlet

Servlets

- Servlets
  - doGet, doPost
  - HttpServletRequest, HttpServletResponse objects
  - Concurrency

Servlets

1. What advantages do Servlets provide over using in-built Java networking classes such as URLConnection?
2. What are some of the mechanisms available for passing parameters to a Servlet?
3. Identify what is wrong/missing in <some_piece_of_Servlet_code>
4. Show web.xml to satisfy some requirement related to Servlet configuration
5. What are differences between Cookies and URL rewriting methods for session tracking?
6. What does a specific Cookie attribute mean?

Software Architecture

Layered Architecture

- Main Concepts
  - Splitting your code into layers
    - Controllers and Services (Spring)
    - Resources and Services (JAX-RS)
  - Developing against an Interface
  - Concept of dependencies
  - Dependency Injection
    - Setter injection
    - Constructor injection

Layered Architecture

1. What are the advantages of developing against an Interface?
2. How does setter injection work?
3. How does constructor injection work

Frameworks

Spring

- Framework that easies writing of web applications
- DispatcherServlet
- Annotation-based
  - @Controller, @Service, @PathVariable
- Beans
  - Java classes that satisfy certain criteria
    - Have getter and setter methods for the properties
    - Setter method names follow a specific format

Spring

1. What are different components within a Spring-based application setup?
2. What is the purpose of servletContext.xml file?
3. Write a Spring Controller for some given requirement
4. Write a unit test for a given Spring service method
5. Use constructor injection to setup a Spring controller with a Spring bean
6. Use setter injection to setup a Spring controller with a Spring bean

REST and RESTEasy

- Framework that easies writing of REST services
- REST architecture principles
- Annotation-based
  - @Request, @RequestBody

REST and RESTEasy

1. What is the uniform constrained interface in REST design
2. Identify whether following methods are idempotent or not
3. Identify whether following sequence of methods is idempotent or not
4. In RESTEasy, describe what is the ApplicationClass

Software Quality

Unit Testing

- What is it?
  - Testing logical units of code
    - Comes down to testing specific methods
- How to do?
  - Identity “code under test”
  - Identify the dependencies
  - Create mock dependencies
  - Set expectations
  - Invoke the “code under test”
  - Assert output is as expected
  - Verify that certain methods were called/not called

Unit Testing

1. Given a method definition, identify all its dependencies from unit testing point of view
2. Given a method definition, write unit test(s) for it
3. Given a piece of code, refactor it to enable writing of unit tests for it

Functional testing

- What is difference between unit testing and functional testing?
- Write a functional test for a particular REST resource

Tools

Marshalling/Unmarshalling

- Marshalling
  - Converting Java objects to JSON/XML
- Unmarshalling
  - Converting JSON/XML strings to Java objects

XML/HTML Parsing

- Parsing techniques
  - XML
    - DOM Parsing
    - SAX Parsing
    - XPath
    - Java regular expression parsing
  - HTML
    - Jsoup

XML/HTML Parsing

Given a XML/HTML response, write code to parse it using <a_parsing_method>