SENG2050 - Lab 01

February 26, 2018

1 Tomcat

1. Download Tomcat 8.5
   - Navigate to `http://tomcat.apache.org/download-80.cgi`
   - Under 8.5.X > Binary Distributions > Core Click (zip).
2. Extract the folder to your U drive.
3. Add the following environment variables:
   - JAVA_HOME=C:\Program Files\Java\jdk1.8.0_X < You need to find X through the file explorer!
   - CLASSPATH=.;PATH_TO_TOMCAT\lib\servlet-api.jar
4. By following these steps:
   - (a) Start
   - (b) Search "Accounts"
   - (c) Click User Accounts control panel
   - (d) In the left pane click Change my environment variables
   - (e) Click New
5. Check correct values by typing in command prompt:
   - echo %JAVA_HOME%
   - echo %CLASSPATH%
6. Navigate to the bin directory in the Tomcat folder.
7. Open a command window in the bin directory (shift + right mouse click then open command window here).
8. Type startup.bat to start Tomcat.
9. Open Edge and navigate to http://localhost:8080 to view Tomcat's default web page.
   - NOTE: When using Chrome or FireFox, accessing the task will cause a DNS error.

2 Basic Servlets

Once your Web Server is working, the next step is start to developing web pages and practice the elements studied in class. The main objective of this part of the tutorial is to introduce you to Servlets. If you are familiar with Java code then you can already know all that is needed to write a Java Servlet.

Basic servlet structure:

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;

@WebServlet(urlPatterns = {"/MyServlet"})
public class MyServlet extends HttpServlet {
    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
                // Use "request" to read incoming HTTP headers
                //(e.g.cookies) and HTML form data (e.g. data the user
                //entered and submitted)
                // Use "response" to specify the HTTP response line
                //and headers (e.g. specifying the content type,
                //setting cookies).
                PrintWriter out = response.getWriter();
                // Use "out" to send content to browser
        }
}
```

To create a Servlet you need to extend HttpServlet and then override the doGet() and/or the doPost() methods. These methods will be used to process the request depending on the browser request method (GET or POST). Read the comments above about the HttpServletRequest and HttpServletResponse objects.

2.1 Simple Text Servlet

In this section, we will examine the code required to display a simple message in the browser.

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import javax.servlet.annotation.WebServlet;

@WebServlet(urlPatterns = {"/SimpleMessage"})
public class SimpleMessage extends HttpServlet {
    public void doGet(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
                PrintWriter out = response.getWriter();
                out.println("Put a simple message here...");
        }
```

}

1. Place this code in a file called SimpleMessage.java (source can be found in blackboard)
2. Save it in your Tomcat directory under webapps\lab01\WEB-INF\classes
3. Compile the servlet (in the same directory as SimpleMessage.java type javac SimpleMessage.java)
4. Create a file Called web.xml in your Tomcat directory under webapps\lab01\WEB-INF with the following content (source can be found in blackboard):

   -
     ```
     <?xml version=”1.0” encoding=”UTF–8”?>
     <web–app version=”3.1”
         xmlns=”http://xmlns.jcp.org/xml/ns/javaee”
         xmlns:xsi=”http://www.w3.org/2001/XMLSchema–instance”
         xsi:schemaLocation=”http://xmlns.jcp.org/xml/ns/javaee http://xmlns.jcp.org
         <session−config>
             <session−timeout>
                 30
             </session−timeout>
         </session−config>
     </web–app>
     ```
   - NOTE: Do not directly copy this text.
   - Either type it out or use the web.xml file located in the supplied resources.zip.

5. Restart your Tomcat server (ctrl+c in the tomcat terminal and then startup.bat again)
6. In Edge visit http://localhost:8080/lab01/SimpleMessage
   - NOTE: When using Chrome or FireFox, accessing tomcat will result in an error!

3 Valid HTML

Write a Java servlet that will output a valid HTML document with an appropriate heading and a short welcome message. Being able to do this effectively is vital for the first assignment. You may format the output however you like, but remember it should be an HTML page. To ensure the validity of your output:

1. Visit your servlet in a browser;
2. Right click and select “view source” or “view page source”;
3. Copy the contents and paste it in the “direct input” textbox found http://validator.w3.org/#validate-by-input .

Hint: the HTML5 doctype is <!DOCTYPE html>