SENG2050 Week 3 Part A

March 2022

In this laboratory session we will learn how to:

- Setup Apache Tomcat;
- Create our first Servlet; and
- Generate valid HTML.

Apache Tomcat is a Java Servlet Container, a type of web server which provides an environment for our servlets to run. When Tomcat detects a HTTP request (such as accessing a page), it will delegate the handling of the request to our Servlet, which can then generate a dynamic HTML page. A Servlet is nothing more than a Java class that handles a HTTP request. This lab exercise assumes Java 11 and Tomcat 9.

1 Apache Tomcat

First, we will download Apache Tomcat from http://tomcat.apache.org. Click Download Tomcat 9.0. Download the latest Binary Distribution > core > zip. Extract this zip file to your U: drive.

2 Environment Variables

In order for Tomcat to run and to enable the compiling of our Servlets we need to set two environment variables. An environment variable is just a "configuration value" that can be used by Tomcat and the Java compiler at runtime.

The first environment variable is JAVA_HOME. This will tell Tomcat where the JDK is installed. The second is CLASSPATH, which will tell Java where to look for classes to compile against. Servlets are not part of a standard Java installation so we need to tell Java where to look for them.

On a lab PC you will need to set these every session!

2.0.1 On Windows 10

First, we need to find the path to our Java installation. On a Lab PC it should be in C:\Program Files\Java\openJDK\<version> (e.g. 11.0.2). Copy this path from the file explorer in preparation for the next step.

Now we can set the values.

1. Click Start > Search for "Environment Variables" > Click "Edit Environment Variables for your account"
2. Under "User variables for <name>", click New
3. The environment variables we need to add are:
    - JAVA_HOME = <path to the Java installation>
    - CLASSPATH = .;<path to tomcat>\lib\servlet-api.jar

Add these variables one at a time (you don't need the equals sign, just split them up). Click Ok when done and it should appear in the list.

2.0.2 On macOS 10.15 or later

Note: This is assuming you are using the default Unix shell for macOS 10.15, zsh. If you are still using bash, you will have to change to zsh. See the instructions when you open a new terminal window.

First, we need to open our user environment profile file. In a terminal window run:

```
open /System/Applications/TextEdit.app ~/.zprofile
```

A new textedit window will open. Add the following lines to this file:

```
export JAVA_HOME=/Library/Java/Home
export CLASSPATH=.:<path to tomcat>/lib/servlet-api.jar
```

Hint: You can drag the tomcat folder into textedit to avoid manually typing the path.

Save this file and close the textedit window before continuing.

Finally, we need to set tomcat to be executable. Open a new terminal window and run the following command:

```
chmod +x <path-to-tomcat>/bin/catalina.sh
```

2.0.3 On macOS 10.14 or earlier

First, we need to open our user environment profile file. In a terminal window run:

```
open /Applications/TextEdit.app ~/.bash_profile
```

A new textedit window will open. Add the following lines to this file:

```
export JAVA_HOME=/Library/Java/Home
export CLASSPATH=.:<path to tomcat>/lib/servlet-api.jar
```

Hint: You can drag the tomcat folder into textedit to avoid manually typing the path.

Save this file and close the textedit window before continuing.

Finally, we need to set tomcat to be executable. Open a new terminal window and run the following command:

```
chmod +x <path-to-tomcat>/bin/catalina.sh
```

2.1 Validating Environment Variables

Close all terminal/command prompt/powershell windows before continuing.

When the environment variables are set, we need to verify that they are correct.

On Windows, open a new PowerShell window, and type the following commands:

```
echo $env:CLASSPATH
echo $env:JAVA_HOME
```

On everything else, open a new terminal and type the following commands:

```
echo $CLASSPATH
echo $JAVA_HOME
```

Ensure that these values are correct. If not, try setting them again. Make sure the classpath starts with .; (Windows) or .: (Unix). This tells Java to look in the current directory for classes. Without it, weird errors can occur in later labs!

2.2 Running Tomcat

Open a new powershell/terminal window.

Navigate to the tomcat 'bin' directory.

On windows, start tomcat with the command

```
./catalina.bat run
```

On all others, start tomcat with the command

```
./catalina.sh run
```

Tomcat should start running in the current powershell/terminal window. This is the tomcat run log. If there are any errors, they are printed here.

Open Edge or Chrome, and navigate to http://localhost:8080 to view Tomcat's default web page. This page has links to useful resources that may be helpful in future labs!

3 Java 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 Servlet.

Basic servlet structure:

```
// @WebServlet tells tomcat what URL to map the servlet to.
@WebServlet(urlPatterns = {"/BasicServlet"})
public class BasicServlet 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 and 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.

3.1 Simple 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/
              xml/ns/javaee/web-app_3_1.xsd">
          <session-config>
               <session-timeout>
                    30
               </session-timeout>
          </session-config>
     </web-app>
     ```
   - NOTE: Do not copy-and-paste this text. Type it out manually.
5. Restart your Tomcat server (ctrl+c in the tomcat terminal and then run the startup command again)
6. In Edge visit http://localhost:8080/lab01/SimpleMessage
     ```

4 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>

Apache Tomcat®

Tomcat 9 Software Downloads

Welcome to the Apache Tomcat® 9.x software download page. This page provides download links for obtaining the latest version of Tomcat 9.0.x software, as well as links to the archives of older releases.

Unsure which version you need? Specification versions implemented, minimum Java version required and lots more useful information may be found on the 'which version?' page.

Quick Navigation

KEYS | 9.0.58 | Browse | Archives

Release Integrity

You **must** verify the integrity of the downloaded files. We provide OpenPGP signatures for every release file. This signature should be matched against the KEYS file which contains the OpenPGP keys of Tomcat's Release Managers. We also provide `SHA-512` checksums for every release file. After you download the file, you should calculate a checksum for your download, and make sure it is the same as ours.

Mirrors

You are currently using **https://dlcdn.apache.org/**. If you encounter a problem with this mirror, please select another mirror. If all mirrors are failing, there are *backup* mirrors (at the end of the mirrors list) that should be available.

Other mirrors: https://dlcdn.apache.org/ Change

9.0.58

Please see the README file for packaging information. It explains what every distribution contains.

Binary Distributions

- Core:
  - zip (pgp, sha512)
  - tar.gz (pgp, sha512)
  - 32-bit Windows zip (pgp, sha512)
  - 64-bit Windows zip (pgp, sha512)
  - 32-bit/64-bit Windows Service Installer (pgp, sha512)
- Full documentation:
  - tar.gz (pgp, sha512)
- Deployer:
  - zip (pgp, sha512)
  - tar.gz (pgp, sha512)
- Embedded:
  - tar.gz (pgp, sha512)
  - zip (pgp, sha512)

Source Code Distributions

- tar.gz (pgp, sha512)
- zip (pgp, sha512)

Apache Tomcat®

Tomcat 9 Software Downloads

Welcome to the Apache Tomcat® 9.x software download page. This page provides download links for obtaining the latest version of Tomcat 9.0.x software, as well as links to the archives of older releases.

Unsure which version you need? Specification versions implemented, minimum Java version required and lots more useful information may be found on the 'which version?' page.

Quick Navigation

KEYS | 9.0.58 | Browse | Archives

Release Integrity

You **must** verify the integrity of the downloaded files. We provide OpenPGP signatures for every release file. This signature should be matched against the KEYS file which contains the OpenPGP keys of Tomcat's Release Managers. We also provide `SHA-512` checksums for every release file. After you download the file, you should calculate a checksum for your download, and make sure it is the same as ours.

Mirrors

You are currently using **https://dlcdn.apache.org/**. If you encounter a problem with this mirror, please select another mirror. If all mirrors are failing, there are *backup* mirrors (at the end of the mirrors list) that should be available.

Other mirrors: https://dlcdn.apache.org/ Change

9.0.58

Please see the README file for packaging information. It explains what every distribution contains.

Binary Distributions

- Core:
  - zip (pgp, sha512)
  - tar.gz (pgp, sha512)
  - 32-bit Windows zip (pgp, sha512)
  - 64-bit Windows zip (pgp, sha512)
  - 32-bit/64-bit Windows Service Installer (pgp, sha512)
- Full documentation:
  - tar.gz (pgp, sha512)
- Deployer:
  - zip (pgp, sha512)
  - tar.gz (pgp, sha512)
- Embedded:
  - tar.gz (pgp, sha512)
  - zip (pgp, sha512)

Source Code Distributions

- tar.gz (pgp, sha512)
- zip (pgp, sha512)
