

Write a JSP file

By: Shahin Sheidaei

Goal: Count user visits of a page

Hint: JSP implementation is easier than Servlet, no need for web.xml ... so first we begin with a JSP implementation and for comparison we implement the regarding servlets too.

We are just needed to put the jsp file in the application folder and that should be working.

So, create a new folder called jserp2 (or any other name you like) under webapps folder of tomcat installation directory and the jsp file there. Now you can access it through

<http://localhost:8080/jserp2/hello.jsp>

(You can download the hello.jsp from the iat 352 lab webpage from the week 4 <http://sc-tomcat.surrey.sfu.ca/iat352/labs/>)

```
<HTML>
  <HEAD>
    <TITLE>Welcome to IAT352</TITLE>
  </HEAD>
  <BODY>
    <H1 ALIGN="center"> Welcome to IAT352</H1>
    <hr align="center" width="80%">
    <p align="center">
      <%
        String name = (String)session.getAttribute("name");
        if (name == null){
          name = request.getParameter("name");
          if (name == null || name.trim().length() < 2){
            name = null;
          }else{
            session.setAttribute("name", name);
            session.setAttribute("time", new Integer(1));
          }
        }
        if (name == null){
      %>
      <form action="hello.jsp" method="GET">
        Please enter your name: <input type="text" name="name" value="" />
        <input type="submit" name="" value="Save My Name">
      </form>
      <%
        }else{
          Integer time = (Integer)session.getAttribute("time");
          int value;
```

```

    if (time == null){
        value = 1;
    }else{
        value = time.intValue() + 1;
    }
    session.setAttribute("time", new Integer(time.intValue() + 1));
%>
<h3>Hello <%= name %></h3>
You access this page <%= time.toString() %> times.
<%
}
%>
</p>
</BODY>
</HTML>

```

The jsp file looks like above, the basic idea is to check if it is the first time you are visiting the page or not, if it is the first time (name==null) then it shows you a box to put your name within it and then each time you refresh the page after submitting your name it shows your number of visits. (What method is it using ? Session? Cookies?)

For better understanding we are going to implement the same functionality using two different servlets, one using cookie (HelloWorldCookie) and the other one using session (HelloWorld) for handling the requests of users.

The .java files are within the package that you downloaded from iat 352 lab page. Like the previous lab we just need to compile the servlets and put them into tomcat.

Follow these steps:

1. Create a new project in netbeans
2. Create a new java class (what the name of your java class should be?)
3. Do not create a main class for the servlet (otherwise you are going to have problems)
4. Copy the content of HelloWorld.java into your new java file.

You might see some problems, like HttpServlet is not imported into your project, you need to download iat 352 course project zip file (from <http://sc-tomcat.surrey.sfu.ca/iat352/>), extract it anywhere you want and in the \iat352\lib you need to point to servlet-api.jar file.

Right click on your project, properties, libraries, Add Jar/Folder and then choose only servlet-api.jar. You may encounter other problems within your code, it might be minor naming problems such as the name of a file and java class is not the same as the class name or other problems, they can be easily fixed.

Do these steps once more for the HelloWorldCookie.java

Now right click on the files and click compile file (F9).

So we generate the .class files by compiling the .java files. They can be found in the build folder of our project. (try locating them)

Now create a directory structure as follow (anywhere other than your tomcat webapps directory)

- SERVLETNAME
 - WEB-INF
 - Classes
 - Web.xml

Copy the two .class files into the classes folder. You need to create a web.xml for your servlets.

The web.xml should look like this (you need to create it)

```
<?xml version="1.0" encoding="ISO-8859-1"?>

<web-app xmlns="http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee http://java.sun.com/xml/ns/j2ee/web-
app_2_4.xsd"
  version="2.4">

  <display-name>Testing Plus</display-name>
  <description>
    Testing Plus Description
  </description>

  <!-- JSPC servlet mappings start -->

  <servlet>
    <servlet-name>HelloWorldCookie</servlet-name>
    <servlet-class>HelloWorldCookie</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>HelloWorldCookie</servlet-name>
    <url-pattern>/servlets/HelloWorldCookie</url-pattern>
  </servlet-mapping>

  <servlet>
    <servlet-name>HelloWorld</servlet-name>
    <servlet-class>HelloWorld</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>HelloWorld</servlet-name>
    <url-pattern>/servlets/HelloWorld</url-pattern>
  </servlet-mapping>
```

```
<!-- JSPC servlet mappings end -->
```

```
</web-app>
```

The SERVLETNAME can be anything you like.

Now copy the SERVLETNAME folder to the webapps directory of your tomcat installation. After that, you are able to access your servlets through the addresses:

<http://localhost:8080/SERVLETNAME/servlets/HelloWorld>

And <http://localhost:8080/SERVLETNAME/servlets/HelloWorldCookie>

Try figuring out the different result they give us using cookies and sessions

Hint: cookie and sessions are browser based, changing the browser can easily change them, you might also try to delete cookies. How you can delete sessions?

TroubleShooting: if you have problems using the servlets, you might want to change the action from hello and cookie to servlets/HelloWorld and servlets/HelloWorldCookie in the .java files, compile them again and put them in tomcat directory (line 53 and line 83)