The Hong Kong Polytechnic University
Department of Electronic and Information Engineering

EIE3320 Tutorial 7: Java Web Programming (Servlets and JSP)

(Deadline for Submission: Check the course information)

Introduction

The following procedures assume that you do not have the administrative right but you can write to C:\temp. Remember to compress the folder C:\Temp\EIE3320 and copy the compressed file to your flash memory before leaving the lab. The procedures have been tested under Windows 7. Tomcat and Eclipse will also work on Mac OS and Linux, but you have to download the programs by yourself.

Downloading and Installing Software

1. Create a folder named "C:\Temp\EIE3320"
   Create a folder named "H:\Temp\EIE3320\Workspace" under your home folder in CF105/CF503
   Create a folder named "C:\Temp\EIE3320\MySQL"
   Visit http://www.eie.polyu.edu.hk/~enhylin/Servlet.html and you should see several files there.

2. Download Java Development Kit from (Skip this step if you have already installed Java JDK)
   http://www.eie.polyu.edu.hk/~enhylin/Servlet.html and unzip the file to C:\Temp\EIE3320.

3. Download Apache Tomcat 7 from https://tomcat.apache.org/download-70.cgi and unzip the file to C:\Temp\EIE3320.

4. Download Eclipse Helios IDE for Java EE Developer from (Skip this step if you work in CF105)
   http://www.eclipse.org/downloads/packages/release/helios/sr and unzip the file to C:\Temp\EIE3320.

5. Download MySQL from http://www.eie.polyu.edu.hk/~enhylin/Servlet.html and unzip the file to C:\Temp\EIE3320\MySQL.

6. Download MySQL Connector/J from http://www.eie.polyu.edu.hk/~enhylin/Servlet.html and unzip the file to C:\Temp\EIE3320\MySQL.

7. Download a SQL script from http://www.eie.polyu.edu.hk/~enhylin/Servlet.html.

8. Add the MySQL JDBC driver to Tomcat by copying "C:\Temp\EIE3320\MySQL\mysql-connector-java-5.1.14\mysql-connector-java-5.1.14-bin.jar" to "C:\Temp\EIE3320\apache-tomcat-7.0.23\lib". This .jar file contains classes that allow Java programs to manipulate MySQL databases. A JAR file (Java Archive File) is a compressed file that contains a number of .class files in a package.

Setting up MySQL Accounts

9. Open a Command Window (Start→Run→cmd) and execute the following command.

    ```
    cd C:\Temp\EIE3320\MySQL\mysql-5.5.8-win32\bin
    ```

    Then, execute the following command in the Command Window to start the MySQL server:

    ```
    mysqld --verbose
    ```

    Make sure that "mysqld.exe" is running by opening up the Windows Task Manager (Start→Run→taskmgr). If not, it is likely that your computer is running a MySQL service on Port 3306. If you have admin right, stop the service. If you do not have admin right, change your MySQL server to use another unused port, e.g., 3307. For details, see http://www.eie.polyu.edu.hk/~enhylin/Files/mysqld_changeport.docx

10. Open another Command Window. Change directory to MySQL and start the MySQL monitor.

    ```
    cd   C:\Temp\EIE3320\MySQL\mysql-5.5.8-win32\bin
    C:
    mysql   -h  localhost   -u   root
    ```

    "root" is the default user account that is created during installation. The –h option is used to specify the host that runs mysqld.exe.

11. At the mysql> prompt, type

    ```
    USE  mysql;
    ```

    Note that mysql is a built-in database.

12. Add the user "guest" to the mysql build-in database. You may copy-and-paste the following command to your Command Windows running mysql.exe.

    ```
    create user 'guest'@'localhost' identified by 'guest';
    grant select, insert, update, delete, create, drop, references,
    execute on *.* to 'guest'@'localhost';
    exit;
    ```

13. Shutdown mysqld.

    ```
    mysqladmin   -h   localhost   -u   root   shutdown
    ```

    You will notice that mysqld.exe running in another command window is terminated.

Creating a Book Database

14. cd to C:\Temp\EIE3320\MySQL\mysql-5.5.8-win32\bin and restart the MySQL server

    ```
    mysqld          --verbose.
    ```

    Then, start the MySQL monitor (client) by typing

    ```
    mysql  -h  localhost  -u  guest  -p
    ```

    When prompt for entering password, type guest.

15. Use WordPad to read C:\Temp\EIE3320\MySQL\Books.sql. Create the Book database by typing

    ```
    source ../../Books.sql;
    ```

    Note that the semicolon is very important. This command will create a database called "books" in C:\Temp\EIE3320\MySQL\mysql-5.5.8-win32\data\books. Note also the use of forward slash '/' for separating directories and filenames in a path.

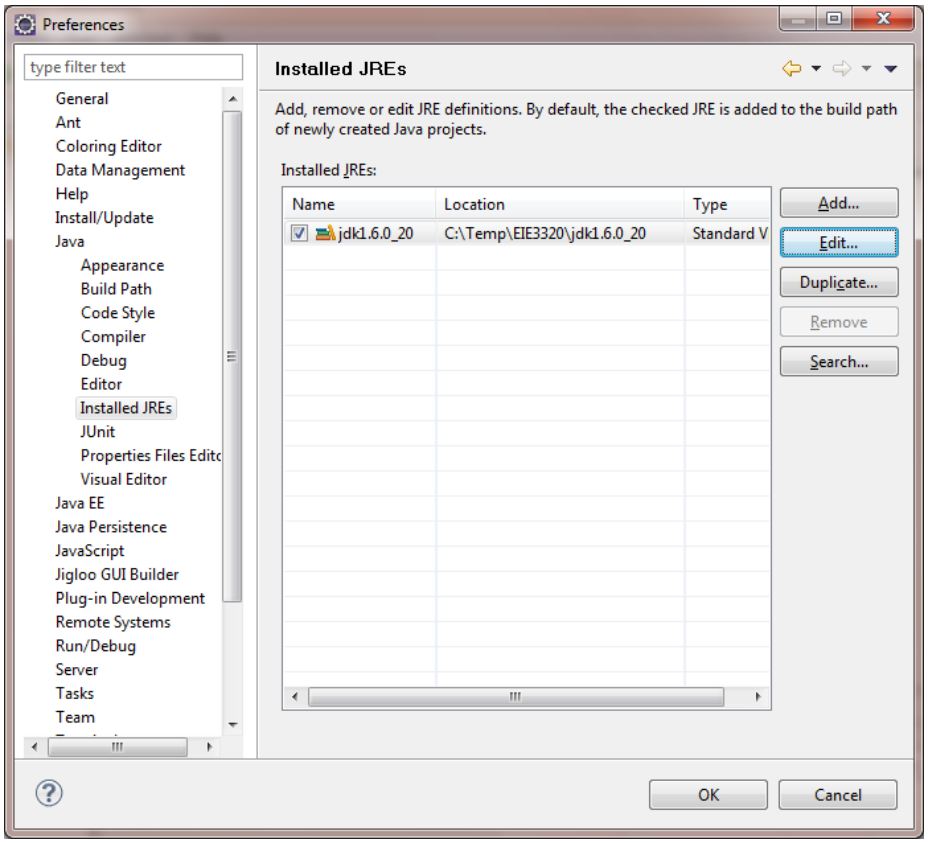
16. Make sure that the database has been successfully created by typing

    ```
    select  *  from authors;
    exit;
    ```

Setting up Eclipse

17. Start Eclipse by clicking eclipse.exe in C:\Temp\EIE3320\eclipse. If it is the first time you start eclipse, it will prompt you to select a workspace. You may select C:\Temp\EIE3320\Workspace as the workspace. When you see Eclipse's welcome page, click "Workbench" in the top-right corner. You may also like to go through the tutorials (if you have time).

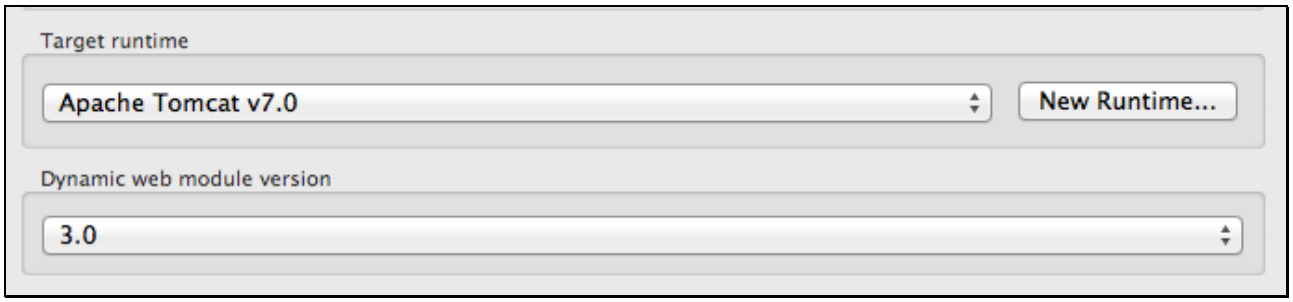
18. You need to tell Eclipse which version of JRE Eclipse will be using. Select Window → Preferences. On the left panel of the Preferences Window, expand Java and select "Installed JRE". Press the "Add" button, select Standard VM, press "Next", press "Directory" and browse to C:\Temp\EIE3320\jdk1.6.0_xx, press "OK" and then "Finish". Then, in the "Installed JREs" Window select jdk1.6.0_xx in the checkboxes. Finally, press "OK".



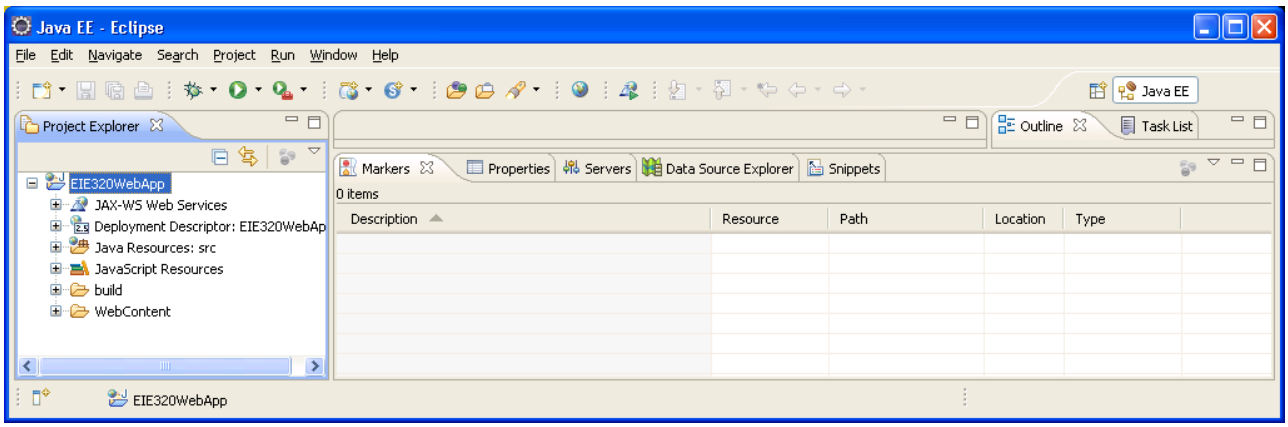
19. Setup Web Application Server. Select Window → Preferences. In the left panel, expand "Server" then select "Run Time Environment". Press "Add" and follow the instruction to select Apache Tomcat 7.0. Select C:\Temp\EIE3320\apache-tomcat-7.0.23 as the installation directory of Tomcat.

Create Web Project in Eclipse

20. Create a new Dynamic Web project called "EIE3320WebApp" using File → New → Dynamic Web Project. Select "3.0" for the Dynamic web module version.



Press the "Next" button to use the default settings. Then, press "Finish".

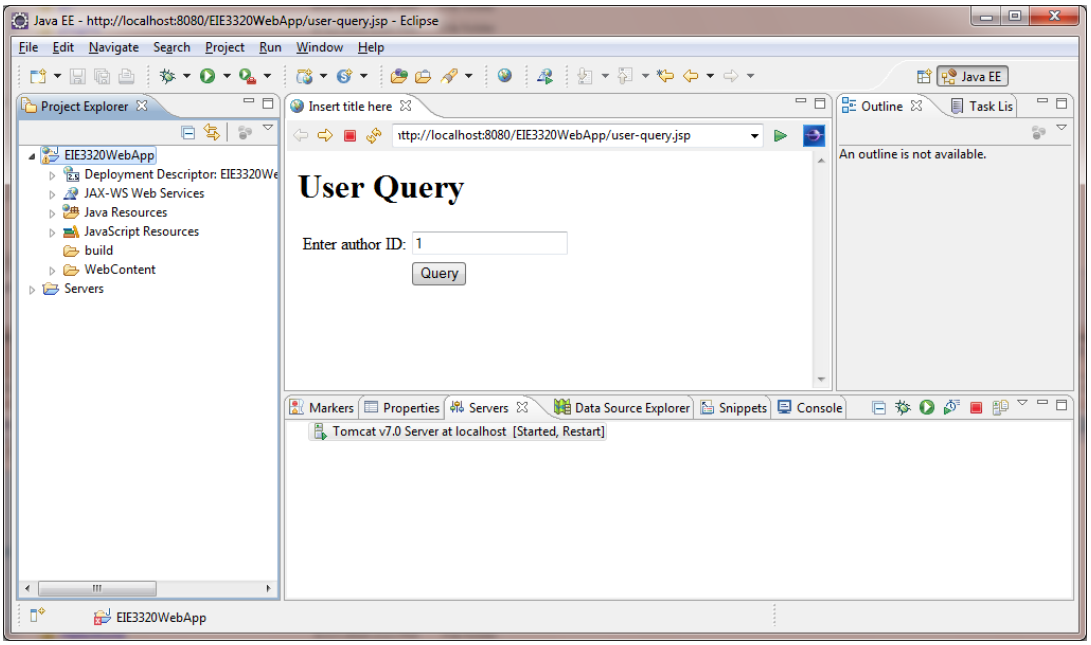


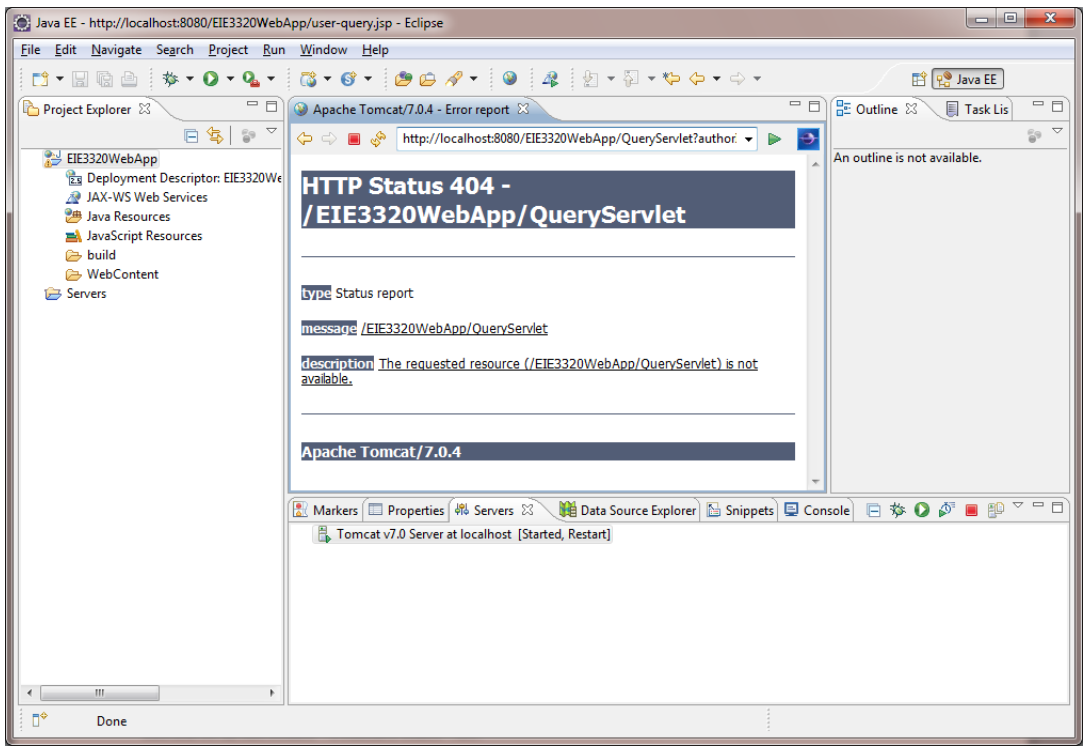
21. Create a JSP file called "user-query.jsp". Right click the project title "EIE3320WebApp" in the Project Explorer on the left and select New → JSP File. Add the following code to the `<body></body>` element of `user-query.jsp`. Save the file. This JSP file will send the user request to QueryServlet.class that runs on the server with the parameter `authorID`. When the server sends back a response, the JSP file retrieves the error object via the session attribute "`error`".

```
<h1>User Query</h1>
<%
   String error_message = "";
   Object error = request.getAttribute("error");
   if (error != null) error_message = error.toString();
%>
<form action="QueryServlet">
<table cellspacing="4">
   <tr>
        <td>Enter author ID:</td>
        <td><input name="authorID" type="text" size="20"></td>
        <td style="color: red"><%= error_message %></td>
   </tr>
   <tr>
        <td></td>
        <td><input type="submit" value="Query"></td>
```

```
        <td></td>
    </tr>
</table>
</form>
```

22. Execute "`user-query.jsp`" by right clicking the file in the Project Explorer (You may see the file by expanding "EIE3320WebApp" and then "WebContent") and select Run As → Run on Server. Follow the on screen instruction to start Tomcat automatically. Input "1" to the edit box and press "Query". You should see the following screen snapshots. Because you do not have "QueryServlet" yet, you will receive Error code 404.





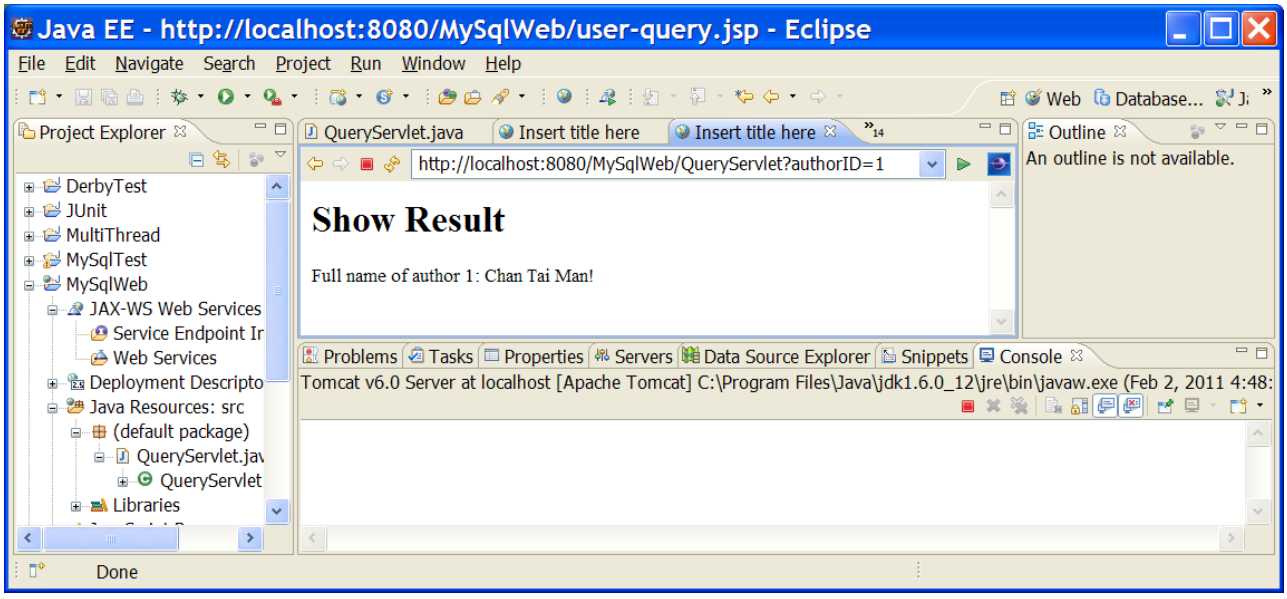
23. Create a new servlet called "QueryServlet". Right click the project → New → Servlet. Use "eie3320" as package name. If Eclipse cannot recognize servlet-related classes (e.g., HttpServlet), you may have done something wrong in earlier steps. In such case, you need to tell Eclipse where to find the file "servlet-api.jar". To do that, right-click the project EIE3320WebApp, select Build Path, then Configure Build Path. In the Library tap, press the button "Add External JAR", and then browse to "servlet-api.jar". It should be in the folder <Tomcat Folder>/lib.

24. Execute user-query.jsp again and see what happen. Explain your observation. Select "EIE3320WebApp/WebContent" as the parent folder.

25. Create another JSP file called "show-result.jsp". Add the following code to `<body></body>`

```
<h1> Show Result</h1>
<%
   String person = "?";
   String id = "?";
   Object authorID = request.getAttribute("authorID");
   if (authorID != null)
        id = authorID.toString();
   Object fullname = request.getAttribute("fullname");
   if (fullname != null)
        person = fullname.toString();
%>
Full name of author <%= id %>: <%= person %>!
```

26. Modify "QueryServlet.java" until you can see the following screen snapshots. Replace "Chan Tai Man" by your name. ***Hints*:** You may need to set some attributes in the *request scope* of your servlet and use the classes `ServletContext` and `RequestDispatcher` to forward the control to "show-result.jsp". To understand the difference between attributes in request scope and session scope, see http://stackoverflow.com/questions/4640721/servlets-setattribute-in-httpservletrequest-vs-setattribute-in-httpsession

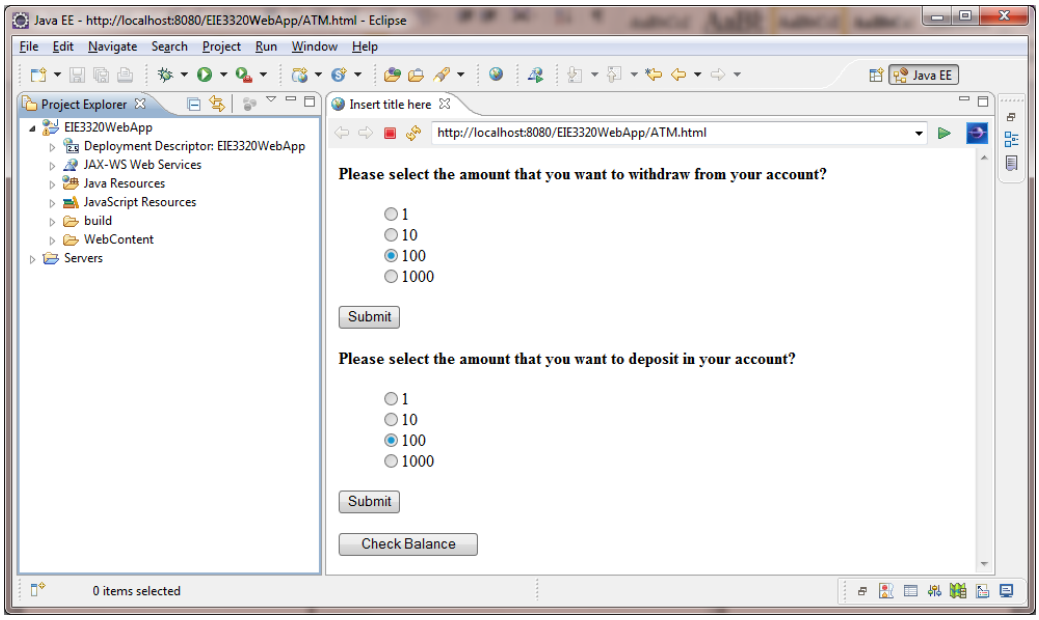


Programming

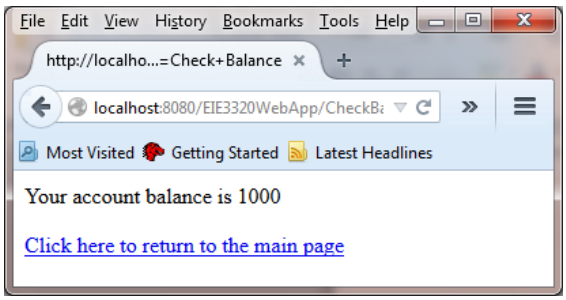
27. **(Assignment)** Create an HTML file called ATM.html and put the following code into the main body.

```
<form action="debit" method="POST">
  <p><strong>Please select the amount that you want to withdraw from your
  account?</strong></p>
  <dl>
    <dd><input type="radio" name="Amount" value="1">1 </dd>
    <dd><input type="radio" name="Amount" value="10">10</dd>
    <dd><input type="radio" checked name="Amount" value="100">100</dd>
    <dd><input type="radio" name="Amount" value="1000">1000</dd>
  </dl>
  <p><input type="submit" value="Submit"> </p>
</form>

<form method="POST" action="credit">
  <p><strong>Please select the amount that you want to deposit in your
account?</strong></p>
  <dl>
    <dd><input type="radio" name="Amount" value="1">1 </dd>
    <dd><input type="radio" name="Amount" value="10">10</dd>
    <dd><input type="radio" checked name="Amount" value="100">100</dd>
    <dd><input type="radio" name="Amount" value="1000">1000</dd>
  </dl>
  <p><input type="submit" value="Submit" name="B1"></p>
</form>

<form method="GET" action="checkBalance">
  <p><input type="submit" value="Check Balance" name="B1"></p>
</form>
```

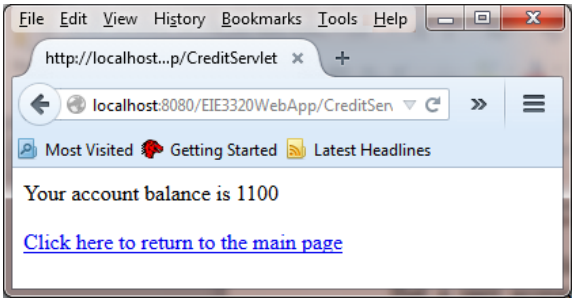
Execute the HTML file by right clicking it and select Run As → Run on Server. You should see the following:



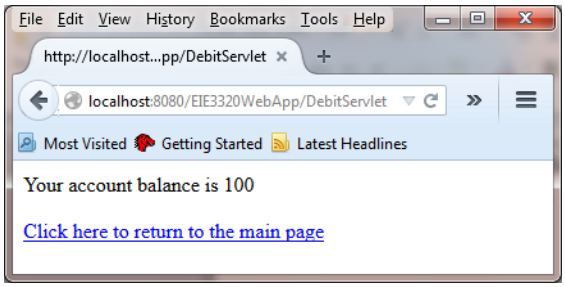
28. **(Assignment)** Create three servlets, namely CreditServlet.java, DebitServer.java, and CheckBalanceServlet.java, so that a user accessing ATM.html can credit money, debit money, and check the balance of his/her account. Note that the servlets need to share a session-based object to keep track o the balance in the account.



After clicking "Check Balance"



After crediting $100



After debiting $1000

Remember to compress the folder C:\Temp\EIE3320 and copy the compressed file to your flash memory before leaving the lab.

Technical Tricks:

1. If you encounter "HTTP Status 404 Error", chances are that Eclipse fails to deploy your webapps to the sever instance folder. In this situation, check whether your .jsp files and .html files can be found in the folder
   Workspace/.metadata/.plugins/org.eclipse.wst.server.core/tmp0/wtpwebapps/EIE3320WebApp/
   To fix the problem, re-save your .jsp, .java, and .html files and re-run the webapps until Eclipse can deploy the tomcat instance to this folder.

Lawrence Cheung
August 2017