

Assignment One: Basic Servlets

Overview:

This assignment is intended to familiarize students with application design, basic servlet functionality, and application deployment with a focus on servlet HTML generation and page navigation.

For this assignment, you will need to create a servlet-based baseball card trade application using servlets to dynamically generate the HTML content using the design elements described below.

Assignment Description:

HTML content must be generated directly and programmatically by the servlets and not by any other method including the redirection to a static HTML pages (unless otherwise noted).

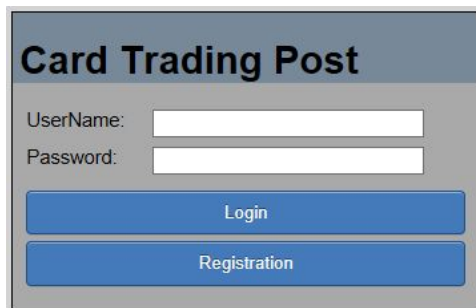
Create a web application that uses servlets to generate web pages that are linked together in order to provide the outline of a complete application that includes:

- A page for user login (html is acceptable)
- A page for new user registration (html is acceptable)
- A page for current baseball collection (servlet) that will also be used to buy and sell
- A placeholder servlet to handle the transaction
- A placeholder servlet to handle registration
- A servlet to handle logout

In this assignment, you are not required to provide any of the application functionality - login, logout, buy cards, etc. - this will all come later. But, you are required to use servlets to generate some of the HTML pages that flow from one page to the next.

A few items to note:

- **login.html**
 - User enters a username/password, clicking the login button on the form should lead to the LoginControllerServlet.java.
 - Clicking the registration button or link should lead to the registration page



The image shows a web form titled "Card Trading Post". It contains two input fields: "UserName:" and "Password:". Below these fields are two blue buttons: "Login" and "Registration".

- **registration.html**

- Create a simple registration form (username, password, confirm).
- Clicking the Register button form should lead to RegistrationControllerServlet.java.
 - Accomplish this as a form action or with javascript.



Trader Registration

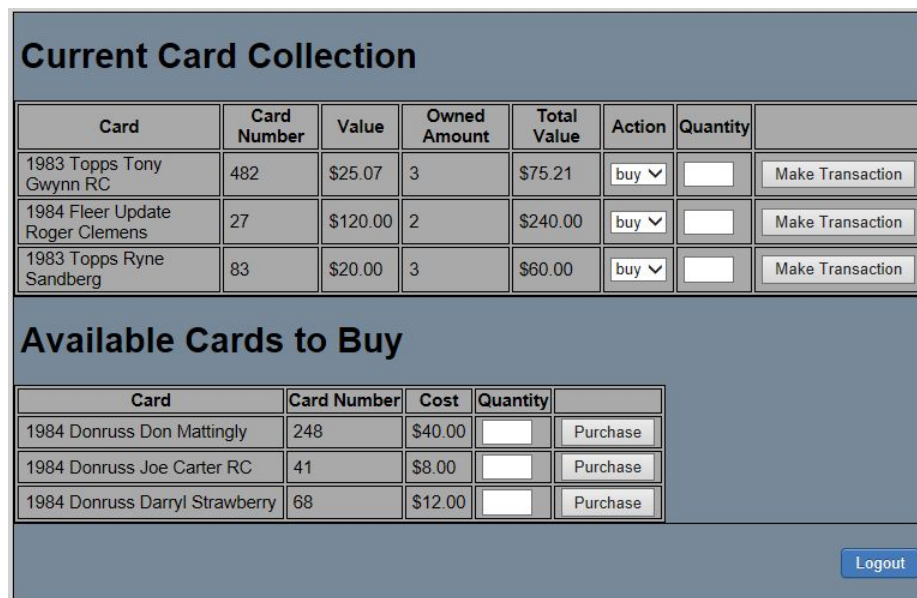
UserName:

Password:

Confirm:

- **CardTradingServlet.java**

- The card collection must list at least two cards and all fields as shown in the accompanying picture
- For this assignment, use placeholder values for the dollar amounts (they can be real or fake).
- The action drop down box on the page listing individual cards should include "Buy" and "Sell".
- Clicking the "Make Transaction" button should lead to the TransactionDetailsServlet.java.
- All fields in the image should be included (order/location replication not required)



Current Card Collection

Card	Card Number	Value	Owned Amount	Total Value	Action	Quantity	
1983 Topps Tony Gwynn RC	482	\$25.07	3	\$75.21	buy ▼	<input type="text"/>	<input type="button" value="Make Transaction"/>
1984 Fleer Update Roger Clemens	27	\$120.00	2	\$240.00	buy ▼	<input type="text"/>	<input type="button" value="Make Transaction"/>
1983 Topps Ryne Sandberg	83	\$20.00	3	\$60.00	buy ▼	<input type="text"/>	<input type="button" value="Make Transaction"/>

Available Cards to Buy

Card	Card Number	Cost	Quantity	
1984 Donruss Don Mattingly	248	\$40.00	<input type="text"/>	<input type="button" value="Purchase"/>
1984 Donruss Joe Carter RC	41	\$8.00	<input type="text"/>	<input type="button" value="Purchase"/>
1984 Donruss Darryl Strawberry	68	\$12.00	<input type="text"/>	<input type="button" value="Purchase"/>

- **TransactionDetailsServlet.java**
 - Provide a summary of the purchase which includes all fields shown in the image.
 - Clicking the "Confirm Transaction" submit button should lead to the TransactionControllerServlet.java which will redirect to CardTradingServlet.java.

Transaction Details					
Card	Number	Buy/Sell	Quantity	Card Price	Total Value
1984 Donruss Don Mattingly	248	Buy	1	\$40.00	\$40.00
Bank Account Information					
Account Holder Name		<input type="text"/>			
Routing Number		<input type="text"/>			
<input type="button" value="Confirm Transaction"/>		<input type="button" value="Cancel"/>		<input type="button" value="Logout"/>	

- **TransactionControllerServlet.java**
 - This is a placeholder.
 - Use a response.sendRedirect to send user to the CardTradingServlet.java
- **RegistrationControllerServlet.java**
 - Placeholder for now. Redirect user to CardTradingServlet.java
- **LoginControllerServlet.java**
 - Placeholder for now. Redirect user to CardTradingServlet.java
- **LogOutControllerServlet.java**
 - Used whenever someone clicks the Logout link / button from any page this servlet is called. For now, redirect user to login page.
































Assignment Submissions:

Your completed application must be loaded and available for public viewing on your AWS server.

What to submit using Blackboard:

1. **HW1.war** - An archive of the entire web application (project) stored in a standard war file. When creating the war file you must include your java source files. This war file will be imported into Eclipse for grading.
 - To create a war file, right click your project > Export > WAR file - **make sure you check Export Source files before clicking finish!!**
2. **info.doc** - Document with the following assignment information:
 - Link to your Login page on your Google server
 - Explanation of status and stopping point, if incomplete.
 - Explanation of additional features, if any.

Workspace Example:

- ▼  assign01
 - >  Deployment Descriptor: assign01
 - >  JAX-WS Web Services
 - ▼  Java Resources
 - ▼  src
 - ▼  constants
 - >  htmlConstants.java
 - ▼  controllers
 - >  LoginControllerServlet.java
 - >  LogOutControllerServlet.java
 - >  RegistrationControllerServlet.java
 - >  TransactionControllerServlet.java
 - ▼  view
 - >  CardTradingServlet.java
 - >  TransactionDetailsServlet.java
 - >  Libraries
 - >  JavaScript Resources
 - >  build
 - ▼  WebContent
 - >  dataFiles
 - >  META-INF
 - ▼  script
 - >  cardCollectionServlet.js
 - >  loginAjax.js
 - >  redirect.js
 - >  registration.js
 - >  transactionDetails.js
 - >  styles
 - >  WEB-INF
 -  login.html
 -  registration.html