

# PROG2240: Java Web Tech

## Assignment 3

Winter 2015

### Due Dates:

Section 1: Monday, 23-Mar-2015, start of class

Section 3: Wednesday, 25-Mar-2015, start of class

Section 2: Thursday, 26-Mar-2015, start of class

---

### Before you begin:

- Read and study sample book applications from Chapters 7, 8 and 9.
- Do the **Individual Exercise 7-2 and 7-3** on page 242. Then, do **Individual Exercise 8-1** on page 267 and demo to your partner. Sign and date **Sections B and C** of your **Pair Programming (PP) Experience Log**.
  - **Note: Your team will get a 50% penalty if the PP Log is incomplete or missing.**
- You must have all the Java source code (i.e., Book.java, BookIO.java) provided in A2.
- Read the Important Notes section of A1.

**Background:** In this assignment, you will continue to use the MVC pattern using servlets and JSPs. You will use EL and JSTL to develop the XXYY Computer Programming Club's **Loan Cart** Web application.

### Important:

- You must **explicitly specify the scope** when accessing attributes using EL
  - You must use <c:url> tag in *href* attribute of an anchor tag (hyperlink).
  - Do not use <c:import> tag when <jsp:include> is adequate.
- 

### Tasks

#### 1. Setup **XXYYClub** NetBeans project

##### a. Add Java classes

- Download **A3Files.zip** and copy these two Java classes to the existing **club.business** package.

- o [ELoan.java](#) and [ECart.java](#)
- Important: **Do not modify any of these classes.**
- [Study all the code provided.](#)
- Use the methods provided as part of your solution (see next steps).

## b. Ensure that the JSTL library is available to this application

---

## 2. Modify the Banner JSP page

- Add two new hyperlinks to the [XXYYBanner.jsp](#).
    - o [eLoan](#) (linked to [XXYYLoan](#) servlet -- see Step 3a)
    - o [eCart](#) (linked to [XXYYCart](#) servlet -- see Step 4a)
  - For the screen layout of these hyperlinks, refer to the sample screen shot in Step 3.
- 

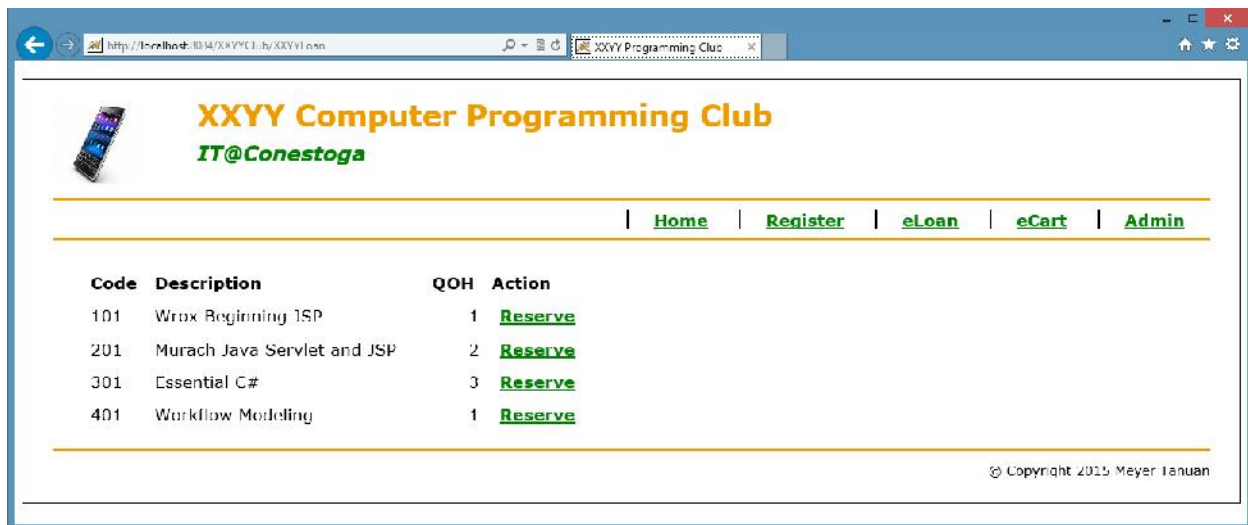
## 3. Write the eLoan servlet and JSP

### a. Add an eLoan servlet

- Under the Java source (i.e., "[Source Packages](#)") directory, add a new package: **club.cart**
- Add a servlet named `XXYYLoanServlet.java`
  - Set URL pattern to [/XXYYLoan](#)
- Within the `processRequest()` generated method:
  - Create a new **servlet context** attribute (i.e., [application](#) object) to save the ArrayList of Book objects available on loan.
  - Maintain a **loanitems** object to store the ArrayList of **Book** objects available on loan.
    - *Hints:*
      - Create/save the **loanitems** object in the **servlet context** object
      - If **loanitems** stored in the **servlet context** object is null, call the **loadItems(string)** method of **ELoan** class. This will populate **loanitems** with the list of books from the books.txt file.
  - Forward the control to [XXYYELoan.jsp](#) page.

## b. Add an eLoan JSP page

- Add a JSP page named **XXYYELoan.jsp**.
- Add the banner and footer pages.
- Using **JSTL** and **EL** (**not** scripting elements), add the code to match the design layout shown below.
- The **eLoan JSP** page shows a list of **loanitems** with the quantity-on-hand (QOH)
- Each time you click the **Reserve** hyperlink, call the **XXYYCart** servlet with two parameters
  - **action**: set to "reserve" .
  - **code**: set to the **code** selected (e.g., **101**).



**Note:** If the **Reserve** link is selected, call the **XXYYCart** servlet (see next step). When items are reserved (i.e., added to the loan cart), the QOH of that loan item is reduced by 1. If the **QOH becomes zero**, the **Reserve** link becomes "N/A" similar to the screen shot shown below (see Step 6, QOH for **code 101**).

## 4. Write the eCart servlet and JSP

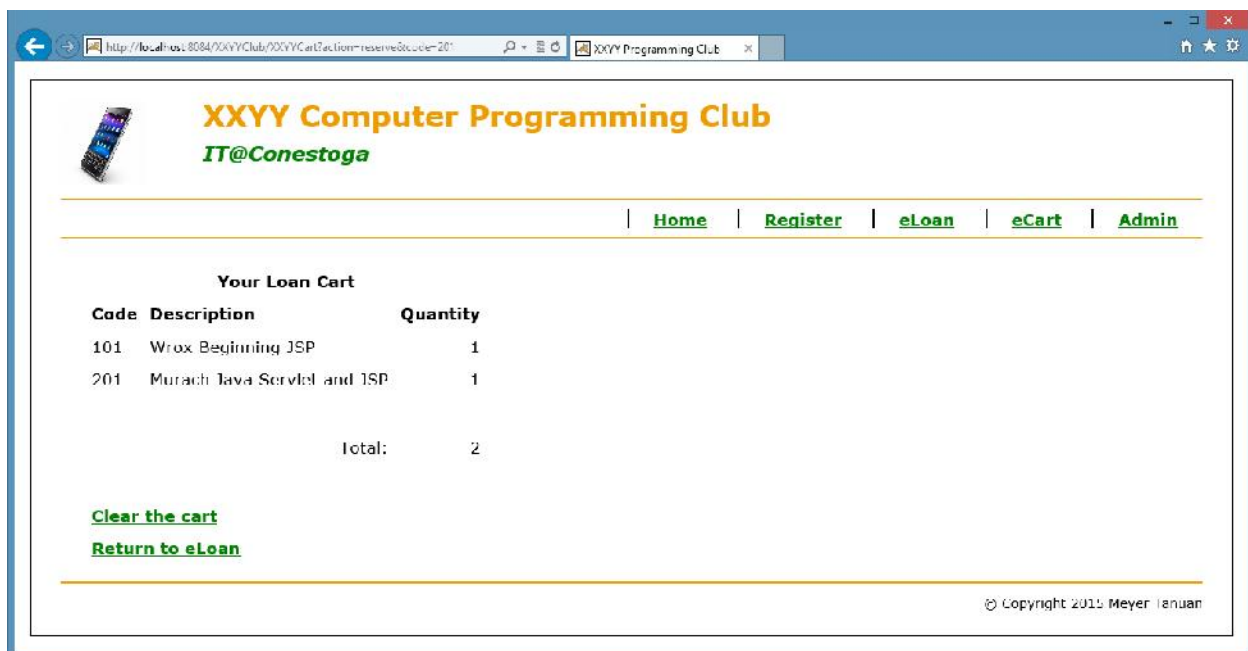
### a. Add the eCart servlet

- Under the **club.cart** package, add a servlet named **XXYYCartServlet.java**
  - Set URL pattern to **/XXYYCart**
- Within the **processRequest()** generated method:

- Access the **loanitems** object stored in the servlet context.
- Create/save a loan **cart** object (instantiated from **ECard** class) in the **session** object
- If action is "**reserve**":
  - Get details of the loan item (i.e., Book object) by calling the **findItem(ArrayList, String)** method of **ELoan** class.
  - Call the **addItem(Book)** method of the loan **cart** object
  - Reduce the loan item's QOH by 1
    - Hint: Call the **subtractFromQOH(ArrayList, String, int)** method of the **ELoan** class.
- Forward the control to **XXYYECart.jsp** page

### b. Add the eCart JSP page

- Add a JSP page named **XXYYECart.jsp**
- Within the <body> tag, include the **XXYYBanner.jsp**
- This page will display all the loan items added to the **cart**.
- Using **JSTL** and **EL** (**not** scripting elements), add the code to match the design layout including the title shown below.
  - Show **total quantity** in the cart (i.e., reserved).
    - Hint: Use the **<c:set>** JSTL tag to declare a variable.
  - **Quantity** column should be aligned to the right.
- Sample screen shot when **Reserve** link is selected for **code 101** and **code 201**:



---

## 5. Write the ClearCart servlet

- Under the **club.cart** package, add a servlet named **XXYYClearCartServlet.java**
    - Set URL pattern to **/XXYYClearCart**
  - Within the **processRequest()** generated method:
    - Create/save a loan **cart** object (instantiated from **ECard** class) in the **session** object, *as appropriate*.
    - Use the **getItems()** method of the **cart** object to access the list of cart items
    - Add back the quantity reserved from each cart item to the loan items QOH.
      - Hint: Call the **addToQOH(String, int)** method in the **ELoan** class.
    - Remove all the cart items in the loan **cart** object
    - Forward the control to **XXYYECart.jsp**
- 

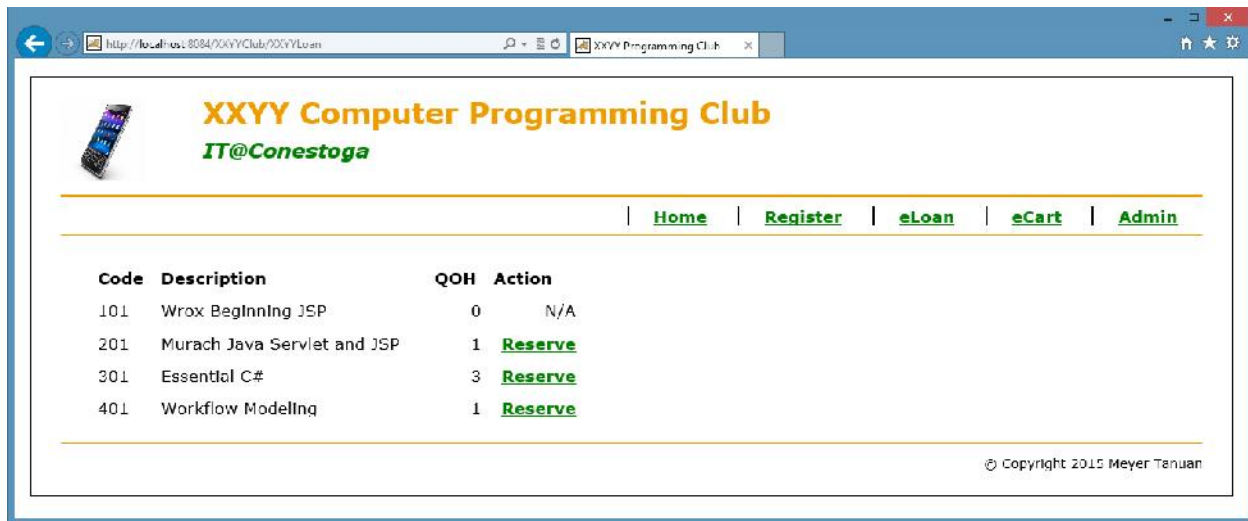
## 6. Continue to Test the Loan Cart Web Application

*Important Note:* If you encounter any errors/issues during testing, you must modify your code to resolve all the errors/issues.

*Before you begin:* Set the Web browser to disable the use of cookies.

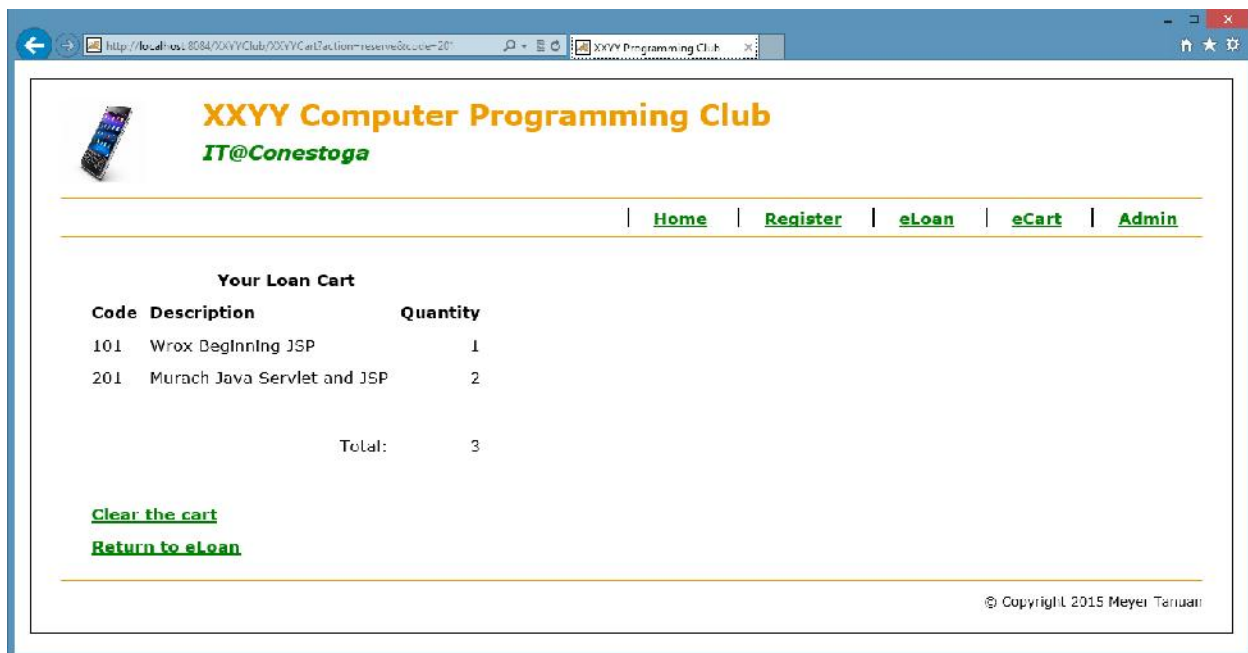
a. After adding code 101 and code 201 to the cart, click the [Return to eLoan](#) link

- Result:
  - **Wrox Beginning JSP** book has been reserved and QOH is **zero**. Action column is shown as "N/A".
  - **Murach Java Servlet and JSP** book has been reserved and QOH is **1**.



b. Click [Reserve](#) link for code 201 again.

- Result: The [Cart](#) is displayed with a total quantity of 3 books in the cart.



c. Click the [Clear the Cart](#) link

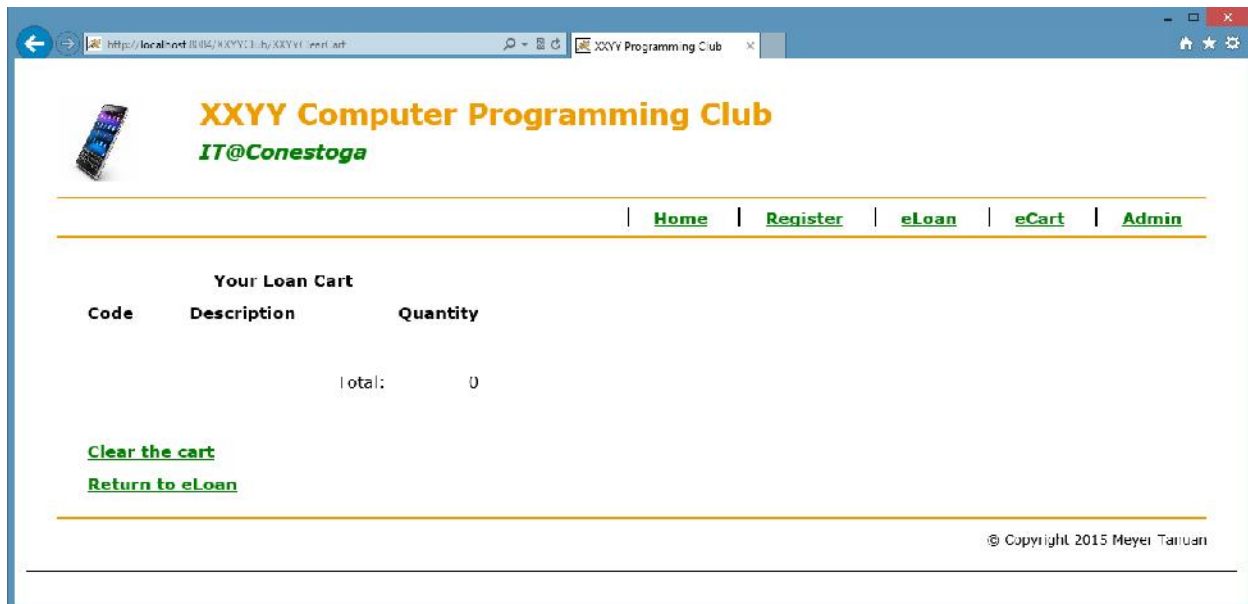
- Result: All cart items are removed from the [cart](#) object. The QOH of the loan items list must be updated accordingly.

d. Click the [Return to eLoan](#) link

- Result: All items are now available to be reserved with QOH shown in sample screen shot of Task 3.

#### e. Testing an Empty Cart

- From the **Banner JSP** page, click on the [eCart](#) link.
  - Result: Empty Cart (see sample screen shot below)
- From the **eCart** page, click on [Clear the Cart](#) link.
  - Result: Empty Cart (see sample screen shot below)



## 7. Testing Multiple Sessions

While the first session (i.e., Web browser) is still open, start another Web browser session.

- Start | All Programs | Internet Explorer (or FireFox)
- Run the **eLoan** page.
  - E.g., <http://localhost:8084/XXYYClub/XXYYLoan>
- In the second session, click **Reserve** link
- **Result:**
  - The QOH must show the loan items that are still available. Loan items reserved by the first (previous) session cannot be reserved by the second (new) session.

- The cart items in the first session are different from the cart items in the second session. This demonstrates the use of the **session** implicit object.

**Tip:** To reset all the QOH to the original quantities, undeploy the **/XXYYClub** application **or restart** the Tomcat server.

---

## 8. Demo your eLoan Web Application

- Print and submit sample screen shots (similar to the five screen shots above) of your running application.
  - Be ready to demo all the features of your assignment solution. Use the above testing steps only as a starting point.
    - E.g., Test when cookies are disabled.
- 

## Assignment Submissions

Note: **You must submit your printouts in the proper sequence.**

Sequence	Item
1	A printout of the <b>CP/CPA Assignment Cover Page and Standards Marking Sheet</b> with all the sections filled in.
2	A printout of the <b>A3 Marking Sheet</b> .
3	A printout of <b>XXYYLoanServlet.java</b> code, where <b>XXYY</b> is your team initials in upper case letters.
4	A printout of <b>XXYYELoan.jsp</b> code.
5	A printout of <b>XXYYCartServlet.java</b> code.
6	A printout of <b>XXYYECart.jsp</b> code.
7	A printout of <b>XXYYClearCartServlet.java</b> code.
8	A printout of screen shots of your running application. Provide the five sample screen shots similar to the above screen shots.
9	A printout of <b>Pair Programming Exercise Log</b> with all the sections filled in.
10	Demo your solution <b>on or before</b> the due date. <b>Both pair programming team members must be present to demo. The instructor will randomly decide who will demo your solution.</b>