

COMP 3512 Assignment #3

Due Monday April 9, 2018 at 10am

Version 1.0, March 20 2018 *[Most recent updated text will be in yellow]*

Overview

You can work in groups of two or three for this assignment. It is also possible to work individually, but I do discourage it; please talk to me about this if you are planning on working by yourself. In this assignment, you will expand your second assignment using JavaScript and jQuery (anywhere this assignment says do some task using JavaScript, you can also use jQuery).

Groups of three will have additional cases to complete. If working in a group, each member needs to take responsibility for and complete an appropriate amount of the project work. **Be sure to consult the instructor at least one week prior to the due date if your group is experiencing serious problems in this regard.**

I feel foolish saying this in a third-year university course, but it is your responsibility to read all the assignment instructions thoroughly and carefully. If you are unclear about something, ask me. But before you do, read the material in question again!

Beginning

You will want to create a new workspace for this assignment. You will likely want to clone your assignment two workspace. Each group member will have to be given RW access to the workspace via the Share button.

Submitting

You will need to share your workspace with me once it is complete. To do so, click the Share button in the upper-right part of the Cloud9 workspace. Invite me via my email (rconnolly@mtroyal.ca) and be sure to give me RW access. As well, you will need to add me as a collaborator to your GitHub repo when you have completed it.

Grading

The grade for this assignment will be broken down as follows:

Usability and visual design	20%
Program design and documentation	15%
Features	65%

USE CASE NAME:	Assignment 2 Functionality
DESCRIPTION:	Expected base functionality
1.	Your system must have ALL the functionality from assignment 2.

USE CASE NAME:	JSON Web Services
DESCRIPTION:	Implement a simple JSON web service
1.	This web service will be a PHP page that simply echoes JSON instead of HTML. I have already provided you with the JSON file <code>printRules.json</code> . Your PHP page simply needs to read it into a string variable using the <code>file_get_contents()</code> function in PHP, set the HTTP Content-Type header to <code>application/json</code> (via the <code>header()</code> function in PHP), and then echo the string.
2.	Name your service <code>print-services.php</code> .

USE CASE NAME:	View Favorites
DESCRIPTION:	Way to view and modify the contents of user's favorites list.
1.	Initiated when user adds an item to the favorites list, or when user clicks any of the view favorites links/buttons.
2.	The system will display a listing of the contents of each of the two favorites list. For favorite posts, display small square image and post title; for favorite images, display small square image and image title. Allow the user to modify the favorites list's contents (i.e., remove an individual item as well as way to quickly remove all items from the list).
3.	For the image favorites list, provide a button labeled "Print Favorites". This button will display a modal dialog box (see Bootstrap documentation) that contains a form that allows the user to "order" prints of each image in the favorites (see Print Favorites Dialog).

USE CASE NAME:	Print Favorites Dialog
DESCRIPTION:	This dialog will use JavaScript and/or jQuery to dynamically calculate and display printing costs dynamically for each image in the favorites list based on user-specified options. It must retrieve the data used for the calculations from the <code>printRules.json</code> data retrieved asynchronously from <code>print-services.php</code> via JQuery. I will make modifications to this file when marking to verify your code is using the data retrieved from <code>print-services.php</code> .
1.	Initiated when user clicks Print Favorites button. This dynamically-generated form should appear in the Bootstrap modal dialog box displayed within the View Favorites form. The basic form structure with each image in the Favorites lists should be populated by PHP. The <code><select></code> lists and calculated values should all be populated by JavaScript/JQuery (JS/JQ). Thus, your page should initiate an AJAX request for <code>print-services.php</code> each time the modal dialog box is displayed.
2.	For each image in the cart, allow the user to select a print size (either 5"x7", 8"x10", 11"x14", or 12"x18") a quantity, a stock and a frame. Defaults are: 5"x7", 1, Matte, None.
3.	Display the possible sizes in a <code><select></code> list. This must be dynamically generated in JS/JQ from the <code>sizes</code> property in the data retrieved from your web service. Thus, when marking, if I change the name of one of the sizes in the <code>printRules.json</code> file, I should see that changed name in the dialog.
4	<p>Display the cost for each image in the cart based on the above values. The base costs for the different sizes are provided in the <code>sizes</code> property in the data retrieved from your web service (for reference, they are \$1.50, \$2.50, \$6.00, and \$7.00).</p> <p>For instance, if the user selected 11"x14" and a quantity of 3, then the total for that image would be \$18 (\$6x3). However, the paper stock and frame selections will change these amounts (see next steps).</p>
5	<p>For each image in the cart, allow the user to select the paper stock in a <code><select></code> list. This must be dynamically generated in JS/JQ using the <code>stock</code> property in the data retrieved from your web service (either Matte, Glossy, or Canvas).</p> <p>The additional cost for the different paper stocks vary depending on the size and are defined by the <code>costs</code> property array in the data retrieved from your web service. For instance, for the Canvas paper stock, the extra costs are \$4 (for 5x7), \$4 (for 8x10), \$8 (for 11x14), \$8 (for 12x18).</p> <p>Thus, if the user selected 11"x14", canvas stock, and quantity of 3, then total for that image would be \$52 ([\$6+\$8]x3). However, the frame selected will change these amounts (see next item).</p>

6	<p>For each image in the cart, allow the user to select a frame in a <select> list. This must be dynamically generated in JS/JQ using the <code>frame</code> property in the data retrieved from your web service.</p> <p>The additional cost for the frame varies depending on the frame type and print size and are defined by the <code>costs</code> property array in the data retrieved from your web service.</p> <p>For instance, the Espresso Walnut frame costs are \$10 (for 5x7), \$12 (for 8x10), \$16 (for 11x14), \$20 (for 12x18). Those costs are for each quantity.</p> <p>Thus, if the user selected 11"x14", canvas stock, quantity of 3, and Espresso Walnut frame, then the total for that image would be \$90 ($[\\$6 + \\$8 + \\$16] \times 3$).</p>
7	<p>For each image in the form, there should be a line total value displayed that is generated by JS/JQ. For instance, if the user had selected 11"x14", canvas stock, quantity of 3, and Espresso Walnut frame for the first image in the favorites list, then the line total of \$90 should be displayed at the end of the line.</p>
8	<p>Display a subtotal (the sum of each line total) and the shipping cost. The shipping cost varies depending upon the shipper and the amount of the order. This must be dynamically generated in JS/JQ using the <code>shipping</code> and <code>freeThresholds</code> properties in the data retrieved from your web service.</p> <p>The shipping options defined by the web service are Standard and Express. Provide a radio button to let the user change the shipping option (the default should be Standard). As with everything else on the form, use the data in the JSON file (for instance, I may change the name of Express in the JSON file to ILOVECOMP3512 and I should see that in the form).</p> <p>The cost for shipping varies depending upon whether there are any frames and whether the total quantity of items is above a certain threshold. If there are no frames in the order, standard shipping is \$5 and express shipping is \$15 (that is a one-time cost regardless of the number). If there are frames and the total frame quantity is less than 10, standard shipping is \$15 and express shipping is \$25; for framed quantities equal to or above 10, standard shipping is \$30 and express shipping is \$45. The values for these rules are defined in the <code>shipping</code> property in the data retrieved from your web service.</p> <p>If the total order before shipping is \$100 or above, then standard shipping is free; if the total order is over \$300, then express shipping is also free. The values for these rules are defined in the <code>freeThresholds</code> property in the data retrieved from your web service.</p>
9	<p>Add a grand total which is subtotal + shipping cost.</p>



8

Add an Order button which submits the user print data via a request to `order.php`. See use case **Submit and Display Order**.

Remember: the display of all the prices should be dynamic. That is, when the user changes any of the <select> items or the shipping radio button, the prices should update immediately (see diagram ... example below isn't styled all that impressively, so try to make it look better).

[Logout](#) [Profile](#) [★ Favorites](#)
Share Your Travels
[Home](#) [About](#) [Browse ▾](#)

Print Favorites

	Size	Paper	Frame	Quantity	Total
	5x7 ▾	Matte ▾	None ▾	<input type="text" value="1"/>	\$1.50
	11x14 ▾	Canvas ▾	Espresso Walnut ▾	<input type="text" value="3"/>	\$90.00

Subtotal \$91.50

☐ Standard ☐ Express *Shipping* \$25.00

Grand Total **\$116.50**

Copyright © 2017 Creative Commons ShareAlike



[Home](#) / [About](#) / [Contact](#) / [Browse](#)

USE CASE NAME:	Submit and Display Order
DESCRIPTION:	Passes user input back to server and displays the result.
1.	Initiated when user clicks the order button in the Print Dialog form.
2.	<p>The form must make a request to <code>order.php</code>. Your form must send the user's form data via POST request. This means each user input element (the <code><select></code> lists and the quantity <code><input></code> element) needs a unique name. The resulting query for the example on the previous page might look like the following:</p> <p><code>size1=0&paper1=0&frame1=0&qty1=1&size2=2&paper2=2&frame2=2&qty2=3&ship=1</code></p> <p>Notice that I am using the <code>id</code> properties for the values selected by the user.</p>
3.	Display the results received in the query string in some nicely formatted table (e.g., example below isn't styled all that impressively, so try to make it look better).

[Logout](#)
[Profile](#)
[Favorites](#)

Share Your Travels
Home
About
Browse

Order Summary

	Size	Paper	Frame	Quantity
	5x7	Matte	None	1
	11x14	Canvas	Espresso Walnut	3

Express Shipping




Copyright © 2017 Creative Commons ShareAlike
[Home](#) / [About](#) / [Contact](#) / [Browse](#)

USE CASE NAME:	Frame Preview
DESCRIPTION:	Show larger version of image with selected frame (only for group of three).
1.	When in the Print Favorites dialog, initiated when the user mouses over the thumbnail image in the form.
2.	<p>Add JavaScript preview of the image when the user mouses over an image in the list. This preview should display the version from the <code>small</code> folder along with the title of the image in a nicely displayed <code><div></code> that is positioned at the location of the mouse pointer. When the mouse moves off the image, then the pop-over preview should disappear.</p> <p>What is different here (compared to the other image preview hover effects from assign 2), is that you will set the <code>style.borderColor</code> and <code>borderWidth</code> style properties of the image preview based on the <code>color</code> and <code>border</code> properties from the frame object in the JSON data retrieved from the web service (see below).</p>

[Logout](#)
[Profile](#)
[Favorites](#)

Share Your Travels
Home
About
Browse

Print Favorites

	Size	Paper	Frame	Quantity	Total
	5x7	Matte	None	1	\$1.50
	11x14	Canvas	Espresso Walnut	3	\$90.00
<div>  </div>					
					<div> <input type="radio"/> Standard <input type="radio"/> Express </div>
					<div> <div>Subtotal</div> <div>\$91.50</div> </div>
					<div> <div>Shipping</div> <div>\$25.00</div> </div>
					<div> <div>Grand Total</div> <div>\$116.50</div> </div>
<input type="button" value="Order"/>					

Copyright © 2017 Creative Commons ShareAlike
[Home](#) / [About](#) / [Contact](#) / [Browse](#)