INFO-1272 JavaScript 1

Final Project

Due Date: April 5th, 2024, 11:59pm

Submission Directions: FOL Submissions folder, Project

Description:

You will be creating a web page simulating an Online Store with the theme of your choice (Flower store, Shoe store, Book store, etc). You have creative freedom over the design of the page layout and experience, so long as it does not interfere with functionality/requirements of the assignment.

The store will feature the following functionality:

- Display catalog of items available (filtered by category)
- Display individual item details based on selection
- Add/remove items to cart
- Calculate order totals (subtotal, tax, shipping, total)
- Support multiple currencies (converted from Canadian dollar)
- Submit reviews and ratings for individual items

Breakdown:

1. Create some constructors for the following Objects:

Store Item *Used for structuring the item data in your online store*	
Id	String – the id for the product
Name	String – the name of the product
Price (Canadian)	Number – the price of the item
Quantity on hand	Number – total amount of item available
Maximum per customer	Number – max quantity allowed in cart
Category	String – the category of the item
Cost of Shipping	Number – the cost of shipping
Reviews	Array – containing review objects
Description	String – the description of the product
Image	String – link to the image file

Cart Item *Used for storing relevant item data specifically for items in the Cart*	
Id	String – the id for the product
Price (Canadian)	Number – the price of the item
Quantity	Number – the amount of this product in the cart
Shipping	Number – the cost of shipping for this item

Review Item *Used for storing a review and rating*	
Review	String – the review of the product
Rating	Number – the rating for the product

2. Create some global variables:
    - An empty array for the store item objects
    - An empty array for the cart item objects

3. Create an initialize function that will be called when the page loads. In this function:
    - Display the current Day/Time to screen
    - Populate the store items array with **at least 15** item objects, with varying data (different categories, quantities, etc)
    - Call your function that will display the store items array *(step 4)*
    - Call your function that will display the cart items array *(step 5)*

4. **Display Store Items:**
   Create a function that will display the store items array in a **dynamically generated table** (or something similar like a card-grid system). This function will be called when the page loads and when the user changes the item category.

   At least the following properties should be displayed:

    - Id
    - Name
    - Price – displayed using the converted currency *(step 7)*
    - Quantity on hand
    - Max per customer
    - Image

5. **Display Cart Items:**
   Create a function that will display the cart items array. This function should be called when the user adds/removes items from the cart.
    - If there is nothing in the cart, output something like “No Items in Cart”

   At least the following properties should be displayed for items in the cart:

    - Id
    - Price
    - Quantity
    - Subtotal (price x quantity)

6. **Create Cart Totals:**

Create a function that will calculate and display the totals for the order in the cart.

- Output the Subtotal of the cart
- Output the estimated shipping
- Output the Subtotal (item subtotal + shipping)
- Output the Tax
- Output the Total (Subtotal + Tax)

These prices should be displayed using the converted currency. *(step 7)*

Note: You can use the tax rate of (13%) even if the selected currency is not Canadian. Shipping cost can just be the sum of each items shipping cost in the cart.

7. **Currency Selection:**

- There should be an element on the page for the user to select their preferred currency
- The selected currency should be displayed/used throughout the webpage. *(Whenever a price is displayed, it is converted to the currency selected when it is displayed)*
- The currencies are: Canadian (default) and an additional currency of your choice
- **Hint:** This could be a function that you call whenever you display a price: convertPrice(price) for example

8. **Add to Cart:**

This can be done in many ways. For example:

- An add button next to each item in the table that adds this item to the cart
- A section of the page where the user enters the item id and quantity they want to add to their cart
- A dropdown for the user to choose the quantity
- Or any other creative way you come up with

When an item is being added to the cart, create a cart item object and populate it with the necessary details from the store item object. Add this cart object to the cart object array.

- If the item (with the same ID) is already in the cart, instead of creating a new Cart Item Object, simply modify the quantity of the existing cart item.
    - If the new quantity exceeds the max per customer for that item, set the quantity to the max amount per customer for that item. (For example: If the quantity is 2 and you're adding 2, but the maximum per customer is 3, set the quantity to 3 not 4)
    - The quantity on hand in the store item object should change when the item is added to the cart. (For example: If I add 2 to the cart, I remove 2 from the quantity on hand for the store item).
- Call your function for displaying cart items
- Call your function for calculating cart pricing

9. **Remove from Cart:**
   Similar to the Add step (above), there should be a way for the user to remove items from the cart.
   - Remove the quantity of that item with the corresponding ID from the cart array.
     - If the quantity is below 0, remove the item entirely from the cart
   - Call your function for displaying cart items
   - Call your function for calculating the cart pricing

10. **Review Items:**
    There should be a way for the user to review a specific item. They can leave either a review, a rating (for example between 1 and 5), or both.
    - Make a review object and add the review to the reviews array for the store item object specified by the user
    - The added review should be displayed (along with any other reviews) when the user views the item details

11. **Validation:**
    When the user is entering any information (add/removing/reviewing items) there should be some validation going on.

    Item Id: *(If the user is entering an item ID on the webpage, the following should be validated)*
    - Validate that a value is entered
    - Validate that the Item ID exists in the inventory

    Item Quantity: *(adding/removing)*
    - Validate the quantity is not less than 1
    - Validate that there is enough quantity on hand for an item when adding it to the cart
    - Validate that the quantity being added to the cart does not exceed the maximum number of items per customer value for the item

12. **Display Item Details:**
    There should be a way for the user to view more item details for an item.
    - In an alert (or an alternate section/way of displaying)
      - Display all store item properties (description, price, etc). You don't need to display images, but you can if you would like.
      - Display the text reviews and the average rating. You do not need to display the rating for each review, but you can if you would like.

Rubric:

Project is worth 25% of your final grade

Marks Available	Marks Awarded For
3	**Coding Style** • Function comments • Internal comments • Proper indentation and use of whitespace • Naming conventions followed for variables, functions, objects, arrays. All representative of what they do
1 **3** **5** **4** **4** **2** **5** **4** **4** **3** **4** **3**	**Functional Code** • Global arrays and variables • Object constructors • Initial loading of page data • Display store items • Display cart items • Calculate cart total • Add item to cart • Remove item from cart • Validation of quantity on add/remove • View item details • Submit item review • Currency conversion
8	**User Experience and Overall Appearance** • Provide a controlled user experience by clearing text fields, changing colours or text fields, setting focus to text fields as required, and outputting input to the page as entered or selected, resetting any select boxes to their default after selection, etc. • Arrangement of page elements, colour themes, front style and size, suitable images • Output is properly formatting with correct use of white space, without spelling and grammatical errors, and a currency format is used for money
Total 53	

Program Submission:

- Zip your web page file and submit the zipped file to the Project Submissions folder. Be sure to test your own submission to ensure that it can be unzipped and run.
- **Name your zipped folder firstName_Project for example Renee_Project**

Submit your project on time!

- Project submissions must be made on time! Late projects will be subject to divisional policy on missed tests and late projects. In accordance with this policy, no late projects will be accepted without prior notification being received by the instructor from the student.

Submit your own work!

- It is considered cheating to submit work done by another student or from another source. Helping another student cheat by sharing your work with them is also not tolerated. Students are encouraged to share ideas and work together on practice exercises, but any code or documentation prepared for a project must be done by the individual student. Penalties for cheating or helping another student cheat may include being assigned zero on the project with even more severe penalties if you are caught cheating more than once. Just submit your own work and benefit from having made the effort on your own!

***If you have taken this course before you must submit a new project!**

Other Information:

- No user of frameworks/libraries (such as: bootstrap, jquery, etc)
- You do not have to use the provided starting code. You are encouraged to structure and style the web page however you like. However, the web page must meet the same functionality detailed in this outline!