PROG2002 Assignment 1

Weight: 30% of your final mark

Due: **Week 3 (Monday, 11.59 pm AEST)**

Please note that this is a group assignment (2 students per group).

Note: Each student must submit their work to the Grade system individually, and the answer should not be exactly the same with your group partner. The idea of group is to provide more opportunity to discuss the assignment with your group partner. Please introduce yourself to the Discussion Board (in the Assessment thread) to find your partner.

Specifications

In this assessment, your task is to develop a dynamic client-site web for a real online book store with a minimum of three pages as follows:

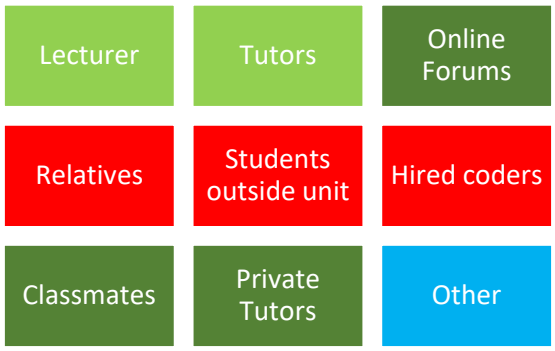
- Page one (i.e., index.html) is a home page containing the book store's general information and a list of available books that customers can buy.
- Page two (i.e., inquiry.html) contains a form to allow customers to send their inquiry (e.g. books that are not available or not in the list) to the book store.
- Page three (i.e., cart.html) displays a list of books selected by the customers (called as a shopping cart) and has a button to send the list to the book store.

The website does not have any database yet hence the list of available books should be stored in an array. The website should contain an appropriate menu on all pages so users can navigate around the website.

Getting Help

This assignment is to be completed individually. It is the opportunity to gain an understanding of the concepts of building a dynamic website. It is important that you master these concepts yourself. You are permitted to work from the examples in the study guide or textbook but you must acknowledge assistance from other textbooks or classmates. In particular, you **must not** use online material or help from others, as this would prevent you from mastering these concepts.

Who can you get help from? Use this diagram to determine from whom you may seek help with your program.



Encouraged

Attribution Required

Ask tutor

Not acceptable

Please note that if your marker has any suspicion that you had help with your code or that your work is not your own you will be asked to come to a meeting with your marker to explain your code. Any student who is unable to explain their code will be submitted for academic misconduct.

Part 1 – Home page

The home page should have the requirements as follows:

- General information about the book store, such as contact info, current promotions, or social pages.
- A list of available books (at least 10 books). Each book has the following information:
  - ISBN (a unique numeric commercial book identifier)
  - Title
  - Authors
  - Price (in AUD)
  - Description
  - Category (e.g. fiction, non-fiction, and children)
- A JavaScript file (e.g. script.js) to store the available books in an array. You need to include the JavaScript file in the home page and display the books on the page. If you do not use an array to store the books, you will not get the full marks.

The page has a menu consisting the following links to allow users to navigate between pages. The menu should be shown in all pages as well.

- Home : a link to go to the home page (index.html)
- Inquiry : a link to go to the inquiry page (inquiry.html)

- Shopping cart : a link to go go the shopping cart page (cart.html)

Part 2 – Inquiry page

The inquiry page contains a form to capture the user's request and send the request to the book store.

The form must contain the following fields:

- Name (required)
- Email (required)
- Phone (optional)
- Message (required)

The form also has a button to send all fields' values inputted by the user.

When the form is submitted, all fields in the form should be validated on the client side using JavaScript according to the following rules:

- All fields must be filled out except the optional fields
- The email must be in a valid format
- The phone should only contain numbers
- The message cannot be longer than 500 characters

If the form validation is not successful, a **bold red** error message should be displayed to the user **on the page** by manipulating the DOM. The message should indicates which fields that have the issues.

If the form validation is successful, a **bold green** success message should be displayed to the user **on the page** by manipulating the DOM. The message should say:

*"Dear **Name**, Thank you for your inquiry. One of our team members will be in touch with you shortly"*

In the above message **Name should be replaced by the user's name from the form**.

Additionally, you must use at least three of the following events in your form: *Onblur, Onchange, Onsubmit, Onfocus and Onreset*.

Part 3 – Home page (continued)

This part aims to improve the home page (Part 1) so it allows users to select books to a shopping cart. The shopping cart consist of a list of selected book with the desired quantity for each book.

The home page should have the following additional requirements as follows:

- For each book displayed on the page, insert a link (i.e. a hyperlink or an image with a link) to allow the user to add the book to the shopping cart with a quantity of 1.
- If the selected book already exists in the shopping cart, the book quantity is incremented by 1.
- Use a web storage (sessionStorage or localStorage) to store a list of selected books including their quantities. You are free to design how you store the information in the web storage.

Part 4 – Shopping cart page

- Page three (i.e., cart.html) displays a list of books selected by the customers (called as a shopping cart) and has a button to send the list to the book store.

The shopping cart page should have the requirements as follows:

- **Display the list of selected books** that have been selected by the customers including their quantity which are stored in web storages (Part 3)
- For each book, there should be **a remove link to delete the book** from the shopping cart.
  In the JavaScript file (e.g. script.js), create a function with a parameter of the book ID. The method aims to delete a book from the web storages based on the book ID.
  Once the remove link is clicked, the method will be called. Once a book is removed from the shopping cart, the user is redirected to the shopping cart page again but with the updated information.
- **A clear link to remove all selected books from the shopping cart** at once.
  In the JavaScript file (e.g. script.js), create a function to delete all the books from the web storages. When the user clicks the clear link, the user should be asked whether they confirm to perform this action via a **JavaScript confirmation**.
  If the user confirms, the method will be called to clear the selected books and then an alert will be displayed saying *"Your shopping cart is now empty"*. The user is then redirected to the shopping cart page again but with an empty shopping cart.
- **A send button to submit the book list in a shopping cart** to the book store.
  When the button is clicked, a validation will be performed to check if there is at least one book in the shopping cart.

If the validation is successful, then display an alert message saying : "*Thank you for your order. We have received it and will process your order soon*", and then the list of books will be removed from the web storages. Hint: You may call the method to remove all selected books.

If it is unsuccessful, display an alert saying "*Your order cannot be processed as your shopping cart is empty. Please select at least one book*"

- You need to **include *the JavaScript file*** *in pages* so you can display the details of selected books. Please note that the web storages only keep the books' ids and their quantities.