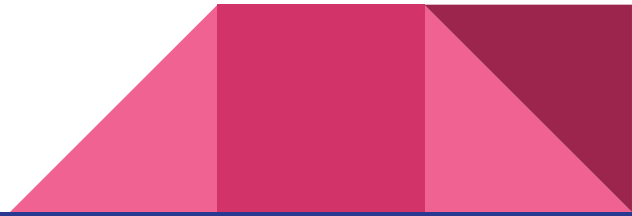
Final Capstone Project

KEALET908_BCL2302_Benjamin_KealebogaLetlhake_IWA19

As a user, I want to view a list of book previews, by title and author, so that I can discover new books to read.

I defined a "Show more" button and set the initial number of books to be displayed to 36. The button displays the number of remaining books to be shown. If there are no more books to show, the button is disabled. When the button is clicked, the next 36 books will be extracted from an array of books and displays them on the page using buttons. The buttons contain information about each book, including the author, title, and image. This process repeats each time the "Show more" button is clicked until all books have been displayed or until the button is disabled.



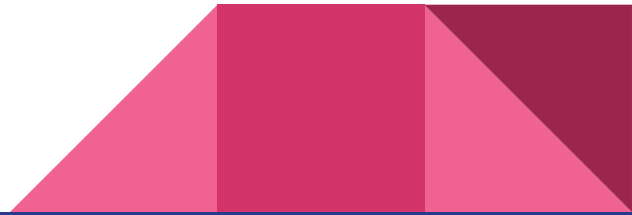
As a user, I want to have the option of reading a summary of the book so that I can decide whether I want to read it.

I defined a function called detailsToggle that is triggered when the user clicks on an element with a class of data-list-items. This function uses a series of conditional statements to check if certain attributes are present on the clicked element and, if so, updates the contents of various HTML elements on the page with the corresponding values. Specifically, the function sets the display property of an element with a class of data-list-active to "block" if the clicked element has a data-id attribute, sets the innerHTML of an element with a class of data-list-description to the value of the clicked element's data-description attribute (if present), and so on for several other attributes.

I also defined an event listener for the click event on an element with a class of data-list-close. When this element is clicked, it sets the display property of the data-list-active element back to "none", effectively hiding the element from view. Finally, I added the detailsToggle function as an event listener for the click event on all elements with a class of data-list-items.

As a user, I want to have the option of seeing the date that a book was published so that I can determine how easy it is to obtain second-hand.

I added an event listener to a book list, represented by the HTML element with the data attribute "data-list-items". When a book in the list is clicked, the detailsToggle function is called, which displays a modal overlay containing information about the book, such as its title, author, description, and image. The information is populated based on the dataset attributes of the clicked book's button element, which are set when the book is added to the list. The event listener is added to the book list, so it applies to all books in the list, both those initially displayed and those added later through a search filter



As a user, I want to filter books by author/genres so that I can find books to read by authors/ genres that I enjoy.

I added a search feature on the web page that allows users to filter and find books based on their titles, authors, and genres. It consists of three parts:

1) the first part handle the search button and cancel button.

2)the second part populates dropdown menus with authors and genres for filtering.

3) The last part to filters books based on user input and display the results.

The first part of the code handles the search button and cancel button. When the search button is clicked, a search overlay is displayed. When the cancel button is clicked, the search overlay is hidden.

The second part of the code creates dropdown menus for filtering books by author and genre. It creates an option for "All Authors" and "All Genres" and then loops through the authors and genres arrays to create options for each one. It then appends these options to the corresponding dropdown menus.

The third part of the code handles the actual filtering of books based on user input. When the search form is submitted, it prevents the default behavior and retrieves the values of the search title, author, and genre. It then loops through the books array and checks if the book matches the search criteria. If it does, it is added to the filteredBooks array. If no books match the search criteria, a message is displayed indicating that no results were found. If at least one book matches the search criteria, the message is cleared and the filteredBooks are displayed.

As a user, I want to toggle between dark and light modes so that I can use the app comfortably at night.

I defined two objects, day and night, which contain two properties each representing a dark and light color. It also selects two elements using querySelector, one with a data-settings-theme attribute and another with a form attribute set to "settings". Then, it attaches a click event listener to the saveButton element which prevents the default form submission behavior.

Inside the click event listener, the value of dataSettingsTheme is checked to determine whether to apply the day or night theme. If dataSettingsTheme is equal to "day", the dark and light color properties of the day object are set as CSS variables using the setProperty method. The same is done for the night object if dataSettingsTheme is equal to "night". Finally, an element with a data-settings-overlay attribute is selected and its display style is set to "none". Overall, this code appears to implement a theme switcher for a website, allowing users to toggle between a light and dark theme.

Day for light mode and Night for dark mode