DWA_08 Discussion Questions

In this module you will continue with your "Book Connect" codebase, and further iterate on your abstractions. You will be required to create an encapsulated abstraction of the book preview by means of a single factory function. If you are up for it you can also encapsulate other aspects of the app into their own abstractions.
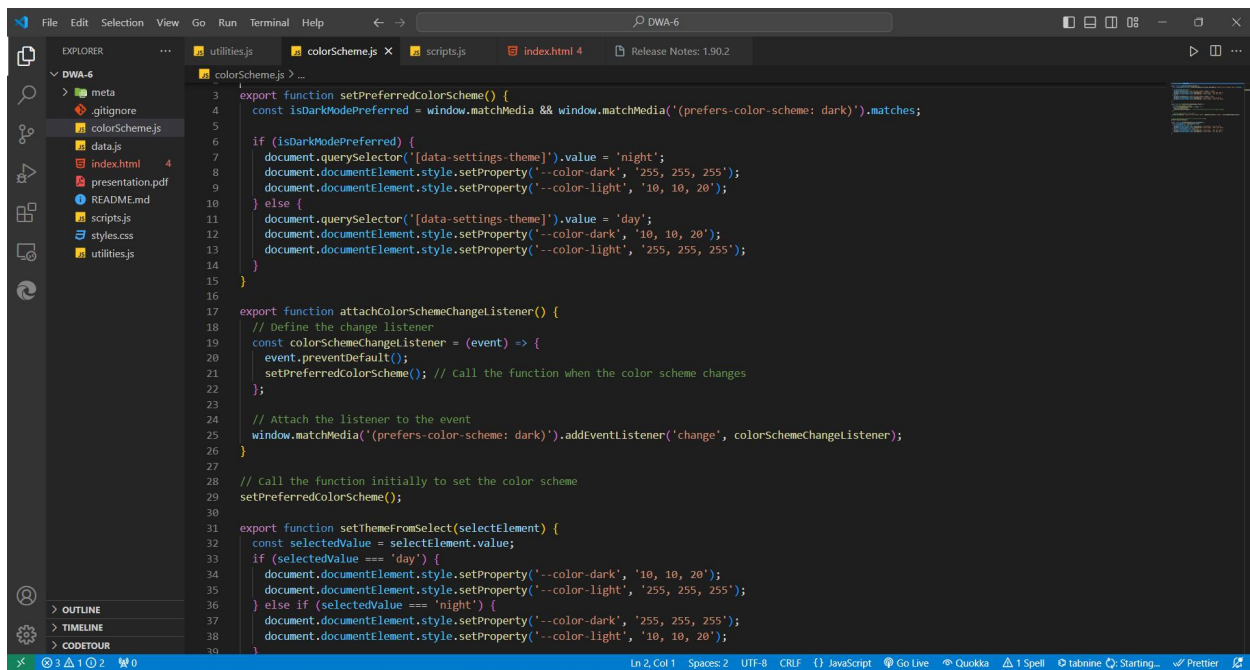
To prepare for your session with your coach, please answer the following questions. Then download this document as a PDF and include it in the repository with your code.

_____

1. What parts of encapsulating your logic were easy?

The parts that are necessary to change mainly being the code block holding the genre
and author option placeholders,and the theme setting.

_____

2. What parts of encapsulating your logic were hard?

The theme setting because i had done it to accommodate the web theme setting
preference

```javascript
export function setPreferredColorScheme() {
  const isDarkModePreferred = window.matchMedia && window.matchMedia('(prefers-color-scheme: dark)').matches;

  if (isDarkModePreferred) {
    document.querySelector('[data-settings-theme]').value = 'night';
    document.documentElement.style.setProperty('--color-dark', '255, 255, 255');
    document.documentElement.style.setProperty('--color-light', '10, 10, 20');
  } else {
    document.querySelector('[data-settings-theme]').value = 'day';
    document.documentElement.style.setProperty('--color-dark', '10, 10, 20');
    document.documentElement.style.setProperty('--color-light', '255, 255, 255');
  }
}

export function attachColorSchemeChangeListener() {
  // Define the change listener
  const colorSchemeChangeListener = (event) => {
    event.preventDefault();
    setPreferredColorScheme(); // Call the function when the color scheme changes
  };

  // Attach the listener to the event
  window.matchMedia('(prefers-color-scheme: dark)').addEventListener('change', colorSchemeChangeListener);
}

// Call the function initially to set the color scheme
setPreferredColorScheme();

export function setThemeFromSelect(selectElement) {
  const selectedValue = selectElement.value;
  if (selectedValue === 'day') {
    document.documentElement.style.setProperty('--color-dark', '10, 10, 20');
    document.documentElement.style.setProperty('--color-light', '255, 255, 255');
  } else if (selectedValue === 'night') {
    document.documentElement.style.setProperty('--color-dark', '255, 255, 255');
    document.documentElement.style.setProperty('--color-light', '10, 10, 20');
  }
}
```

```javascript
import { books, authors, genres, BOOKS_PER_PAGE } from './data.js';
import { setPreferredColorScheme, attachColorSchemeChangeListener, setThemeFromSelect } from './colorScheme.js';
import { createAndAppendAuthorOptions, createAndAppendGenreOptions } from './utilities.js';

const themeSelect = document.querySelector('[data-settings-theme]');
themeSelect.addEventListener('change', () => {
  setThemeFromSelect(themeSelect); // Call the function to update the theme
});

const settingsForm = document.getElementById('settings');

// Add event listener to the form's "submit" event
settingsForm.addEventListener('submit', (event) => {
  event.preventDefault(); // Prevent the form from submitting and refreshing the page
  document.querySelector('[data-settings-overlay]').open = false;
  setThemeFromSelect(themeSelect);
});
// Call the function initially to set the color scheme
setPreferredColorScheme();

// Attach an event listener to update the color scheme dynamically
attachColorSchemeChangeListener();
```

3. Is abstracting the book preview a good or bad idea? Why?

It is a good idea because it makes it easier to change your parts of the code without actually ruining everything and if you are part of a team the risk of someone making a mistake and ruin or damage well functioning code is less.

_____