

# 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?

- Creating book previews: The `createBookPreview` function encapsulates the logic of creating a book preview element based on the provided book object. It takes input parameters, creates the necessary DOM elements, sets their properties, and returns the final preview element. This encapsulation allows for easy reuse and customization of book previews.
- Creating the `createOptionElement` function: This function encapsulates the logic of creating an `<option>` element with specific values and text. It takes input parameters, creates the element, sets its properties, and returns the element. This function encapsulates a specific logic in a reusable manner.

---

## 2. What parts of encapsulating your logic were hard?

- Data filtering and pagination: The code includes logic for filtering books based on user-selected options and implementing pagination to display a limited number of books at a time. It can be complex to encapsulating the filtering and pagination logic, especially when dealing with large data sets and dynamic updates.

- Event handling and DOM manipulation: The code includes various event handlers that manipulate the DOM based on user interactions. It can be challenging to encapsulate the logic related to event handling and DOM manipulation, especially when multiple elements and interactions are involved. Ensuring proper separation of concerns and maintaining the desired behavior can require careful planning and implementation.

---

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

- Abstracting the book preview into a separate component is a good idea. Because by abstracting the book preview, you can encapsulate its functionality, styling, and behavior in a reusable component. This allows for better code organization, modularity, and reusability.
-