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 the factory function that encapsulates the logic of a single preview was fairly simple.
- After following the concepts learned from the lecture videos and reading material, it seemed relatively straightforward.
- It also helped that my logic was already quite well abstracted, so it was easy to extract all the logic relating to the preview and encapsulate it in the factory function.

---

2. What parts of encapsulating your logic were hard?

- In an attempt to further illustrate my understanding of the topic, I tried to refactor my codebase (like what was done in the lecture video) to make effective use of the encapsulation rather than using a global state variable.
- However it proved to be a bit more difficult than anticipated and I could not find a simple, meaningful way of achieving it without making the code seem more complex.

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

- Overall, encapsulating the book preview is a good idea.
- It allows us to manage and better control the data relating to a preview; we can control what data can be accessed or manipulated and the manner in which these actions are performed.
- It also abstracts away the low level details related to instantiating a preview and makes the code more reusable.
- It makes the code more flexible and easy to extend, for example if we would like to add functionality to update the preview's html it would be easier to do because all the logic is encapsulated in a single place.