DWA_09 Discussion Questions

In this challenge, you will continue with your “Book Connect” codebase and further iterate on your abstractions.

Previously, you worked on adding abstraction around the book preview functionality of the project. Next, you must turn the book preview abstraction into a fully-working web component. Then, apply the techniques you’ve learned about this module to the book preview.

If you are up for it, you can also convert other aspects of the app into web components.

You will have a call with your coach where you highlight the following:

---

1. What problems did you encounter converting the book preview to a component?

Firstly, it was dealing with the component Structure, like ensuring that the structure of my web component is accurate and also representing the book preview was rather challenging. I needed to create the necessary HTML structure and styling within your component's shadow DOM to match the original design. And that was a huge challenge to do with so little knowledge on turning a code into a web component.

Secondly, when it came to styling, I had problems with making sure that my component's styling doesn't clash with the styles of the parent application. I thought I may need to use CSS methodologies like BEM (Block, Element, Modifier) or CSS-in-JS to scope my component's styles that I researched about and used CHaTOPEN ai and youtube videos which were very helpful.

---

2. What other elements make sense to convert into web components? Why?

It was the custom buttons since they are a popular choice for web components. Anyone who's coding can create buttons with specific styles, icons, or functionality and reuse them throughout your application since that is how they were used here in the book preview.

And with the use of modal Dialogs that are commonly used for displaying overlays, pop-ups, or modals in the book preview. I have seen that creating a modal dialog as a web component allows you to easily reuse it for different parts of your application.

---

3. Why does keeping your HTML, CSS and JavaScript in a single file sometimes make sense?

I think that is important because of simplicity and readability to the person who doesn't know the code or project it can be easier for the next person to see what is happening within the project because what I have learnt is that having all the code in one file makes it easy to understand the component's structure and behavior when you see it. This can be especially helpful for simple components or prototypes.
