

DWA_07.4 Knowledge Check_DWA7

1. Which were the three best abstractions, and why?

*createPreviewElement Function:

This function encapsulates the creation of a preview element for books, adhering to the Single Responsibility Principle (SRP). It has a clear responsibility: creating a preview element.

It's a cohesive and reusable abstraction that improves code readability.

*setTheme Function:

The setTheme function abstracts the process of setting the theme's CSS variables based on the selected theme. It follows SRP and encapsulates a specific responsibility. It promotes code reuse and maintainability by centralizing theme-related logic.

*loadBooks Function:

The loadBooks function abstracts the process of loading and displaying a specific set of books. It manages the state of the books to display.

It contributes to code organization, and its purpose is well-defined, adhering to SRP.

2. Which were the three worst abstractions, and why?

*handleBookClick Function:

This function appears to handle book click events but lacks a clear single responsibility. It combines event handling, DOM manipulation, and book retrieval.

*closeSearchOverlay and closeSettingsOverlay Functions:

These functions are responsible for closing search and settings overlays but don't follow the Single Responsibility Principle (SRP). They mix UI state manipulation and DOM selection.

*loadMoreBooks Function:

The loadMoreBooks function is responsible for loading more books when the "Show more" button is clicked. While it has a single responsibility, it can be improved for maintainability.

3. How can The three worst abstractions be improved via SOLID principles.

*handleBookClick Function:

To improve this abstraction based on SOLID principles:

Separate event handling from book retrieval and DOM manipulation. Create separate functions for each responsibility.

Use the Open-Closed Principle (OCP) to make it open for extension (e.g., handling more book-related actions) without modifying the existing code.

*closeSearchOverlay and closeSettingsOverlay Functions:

To improve these abstractions based on SOLID principles:

Separate UI state manipulation (opening/closing overlays) from DOM selection and manipulation.

Consider creating a dedicated UI manager class or component that handles UI state.

*loadMoreBooks Function:

To improve this abstraction based on SOLID principles:

Consider separating the logic for loading more books from the logic that updates the "Show more" button's text and disabled state.

Create separate functions for loading books and updating UI elements to enhance readability and adherence to SRP.
