DWA_07.4 Knowledge Check_DWA7

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

Functions: I have used functions to break down the code into smaller, more manageable and reusable pieces of code.

Classes: YES i know i maybe should not have added 1 but i made a Person class to group related data and functions together , This makes it easier to organize and manage complex code.

Control Structures: You have used control structures like if-else statements and for loops to make the code more readable and concise.

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

1) The createBookPreviewHtml function: While it's a good idea to separate the generation of HTML from the rest of the code, this function is tightly coupled to the book object structure. If the book object structure changes in the future, this function will need to be updated as well

2) The createListItems Function : While this function is well-encapsulated, it's doing too much

3) The elements object: While the elements object is a good way to abstract away the querySelector calls, it can be improved by defining the selectors as constants instead of strings.

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

1) The createBookPreviewHTML - A better abstraction would be to define a separate class for the book preview element, which can take in a book object and generate the HTML based on that. This way, if the book object structure changes, only the book preview class needs to be updated.

2) The createListItems Function : It's better to separate out the functionality for generating the HTML for the list of books and the dropdown menus into separate functions. This way, the code is more modular and easier to maintain.

3) The elements object: it can be improved by defining the selectors as constants instead of strings. This way, if the selectors need to be changed in the future, they can be updated in one place (i.e., the constants) instead of scattered throughout the code.
