DWA_01.3 Knowledge Check_DWA1

_____

1. Why is it important to manage complexity in Software?

- Managing complexity in software is important for maintainability, readability, debugging, scalability, team collaboration, and future adaptability. By keeping software systems manageable and understandable, developers can build robust and maintainable applications that meet the needs of stakeholders and users.

_____

2. What are the factors that create complexity in Software?

- Requirements.
- Tight Coupling and Dependencies.
- Time and Resource constraints.
- Team Dynamics and Collaboration.
- Lack of Documentation and Communication.
- External Dependencies and APIs.
- Lack of Modularity and Abstraction.
- Legacy Code and Technical Debt.

_____

3. What are ways in which complexity can be managed in JavaScript?

- Organizing code and Modularization.
- Functional Programming.
- Separation of Concerns
- Code Readability and Consistency
- Testing and Test-Driven Development (TDD).
- Abstraction and Encapsulation.
- Documentation.
- Error Handling and Defensive Programming.
- By applying these techniques, you can effectively manage complexity in your JavaScript code and improve its maintainability, readability, and overall quality

_____

4. Are there implications of not managing complexity on a small scale?

- Yes there are implications of not managing complexity on a small scale by, Reduced Readability and Understanding
- By Increased Bug Occurrence
- The Lack of Scalability
- By Maintaining Burden
- Reduced Long-Term Maintainability
- Decreased Collaboration
- Difficulty in Onboarding New Team Members

It is important to address complexity, regardless of project size, to improve code quality, maintainability, and the overall development experience. By proactively managing complexity, developers can reduce the likelihood of these implications and ensure a smoother development process.

_____

5. List a couple of codified style guide rules, and explain them in detail.

- Camel casing for variable and function names
  * Camel case is a naming convention where multiple words are combined, and each subsequent word starts with a capital letter, except for the first word which starts with a lowercase letter.

- Using block statements, even for single-line conditionals or loops
  * Block statements consist of enclosing curly braces around a group of statements.

- Avoid using `eval()` function
  * The `eval()` function in JavaScript evaluates and executes arbitrary code passed as a string.
- Always use strict equality instead of loose equality
  * In Javascript, the strict equality operator compares both value and type, while the loose equality operator performs type coercion before comparison.

_____

6. To date, what bug has taken you the longest to fix - why did it take so long?

- The bug that took me the longest to fix was using the code exported from the other file into the current file that I am working on. I struggle to know how to use the code as it is already given in the existing file.
- In the previous projects, I had a bug that took the longest to fix, which is making the search button to search for the books that were provided in the other javascript file

_____