

DWA_01.3 Knowledge Check_DWA1

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

1. Maintenance and debugging- When code is simple and well-organized, it's easier to identify and fix issues.
 2. Readability- Code that is easy to read and understand reduces the learning curve for new developers and facilitates collaboration.
 3. Performance: Complexity can negatively impact software performance.
-

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

1. Change- Frequent changes to the software's requirements or design can make it more complex
 2. Dependencies: Relying heavily on other software components or services can add complexity, especially if they are complex or have issues
 3. Messy Code: Poorly organized and unclear code is harder to work with, making the software more complex.
-

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

1. Function Decomposition: Break down large functions into smaller, focused functions that handle specific tasks. This promotes code reuse and makes code easier to read and maintain.
2. Code Comments and Documentation: Add meaningful comments and documentation to your code to explain complex logic, especially when it's not immediately obvious.
3. Namespacing: Use namespaces or objects to group related variables and

functions, preventing global scope pollution.

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

Yes, there are implications of not managing complexity even on a small scale in software development.

1- Reduced Readability: Code that lacks simplicity and structure can be hard to read and understand. Even in a small project, this can make it challenging for you and your team to grasp the code's purpose and functionality.

2- Maintenance Challenges: As your small project grows or evolves, unmanaged complexity can lead to maintenance challenges. Simple updates or additions may become time-consuming and error-prone.

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

1- Rule: Indentation and Whitespace: This rule dictates how you should format and use whitespace and indentation in your code. It ensures consistent and readable code by specifying the number of spaces or tabs for indentation and how to handle whitespace around operators, commas, and other elements.

2-Rule: Naming Conventions: Naming conventions define how variables, functions, classes, and other identifiers should be named. Consistent naming conventions improve code readability and maintainability. Common naming conventions include camelCase, PascalCase, and snake_case.

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

The title filter search for my IWA Capstone project. The code was long and had been
done previously so it was hard to pinpoint where the error was and how to fix

it._____