DWA_01.3 Knowledge Check_DWA1

---

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

Managing complexity in software is important because it ensures maintainability, enables efficient debugging and troubleshooting, facilitates collaboration, improves scalability and performance, promotes code reusability and modularity, and enhances the overall user experience.

---

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

Software complexity can arise from factors such as size, dependencies, interactions, business rules, technical requirements, integration, evolving requirements, concurrency, legacy code, and lack of documentation.

---

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

- Modularize code into smaller, reusable modules.
- Use clear and consistent naming conventions.
- Separate concerns to improve code organization.
- Document code through comments and documentation.
- Regularly refactor to simplify complex sections.
- Adopt test-driven development for early issue detection.
- Collaborate through code reviews and pair programming.
- Apply design patterns for standardized solutions.
- Avoid overly clever code in favour of simplicity.

---

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

Not managing complexity on a small scale can result in difficulties in understanding and maintaining code, increased development time, higher chances of introducing bugs, limited scalability, poor code quality, negative impact on collaboration, and increased technical debt.

---

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

- Use descriptive variable and function names: Emphasizes using meaningful and descriptive names to enhance code readability and understanding.
- Maintain consistent indentation and spacing: Promotes uniformity in code formatting, making it easier to identify code blocks and understand the control flow.
- Limit line length: Sets a maximum line length to prevent excessively long lines that can decrease code readability and require horizontal scrolling.
- Follow naming conventions: Specifies consistent naming conventions for variables, functions, and classes to ensure clarity and consistency in the codebase.
- Avoid unnecessary comments: Encourages writing self-explanatory code and discourages the use of redundant or misleading comments that can become outdated and confuse readers

---

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

During my project on IWA 18 I didn't link my script and HTML proper, so it took me a while to realise that because I was way too focused on my JavaScript codes and not if I linked my things well.

But the most challenging time to debug was when I misspelled my variable and I didn't have a way to debug or know where the mistake was, it took me so long to get the mistake because I was just looking through the code not scrutinizing proper.