DWA_01.3 Knowledge Check_DWA1

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

Because it's not good enough to write code that works. It must be easy to make improvements on the initial code, even for people who weren't involved in the original construction.

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

- Lack of Modularity and Reusability: If software components lack modularity and reusability, it can lead to duplicated code, increased maintenance efforts, and difficulties in managing changes. This lack of modular design and reusable components adds complexity and reduces flexibility.
- Tight Deadlines and Pressure: When there is pressure to deliver software within tight deadlines, it can lead to shortcuts and compromises.

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

- Move repeated Code/nested if else to a separate function
- Ensure all related code is grouped together
  Aim for expressiveness even if the code is longer
- Ensure that data is correct before running

- Add comments to provide additional context
- Using code styles to increase readability, like Use ternary operator instead of if-else loop if you have 2 conditions and spacing or indentation.
- Aiming for modularity and flexibility of code

---

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

- Codebase becomes heavy to comprehend
- When things get too complicated and messy, it wastes time and resources. We end up doing the same things multiple times, using up more time and effort than necessary
- If we don't handle complexity well, it becomes hard to grow and adapt.
- If complexity is not managed properly, it can result in a bad experience for customers.
- It can lead to confusion, inefficiency, mistakes, difficulty adapting, strained teamwork, and unhappy customers.

---

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

Airbnb JavaScript Style Guide

- Group your shorthand properties at the beginning of your object declaration.
  Why? It’s easier to tell which properties are using the shorthand.

- When programmatically building up strings, use template strings instead of concatenation
  Why? Template strings give you a readable, concise syntax with proper newlines and string interpolation features.
- Use dot notation when accessing properties

- Always use const or let to declare variables. Not doing so will result in global variables. We want to avoid polluting the global namespace
- For spacing or Indentation, Use soft tabs (space character) set to 2 spaces

---

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

- It was on IWA17, I was struggling to return a string containing an HTML table cell with the specified properties and value and also, because loops are a bit of a challenge to me.
- IWA 18, where I had to drag and drop the order from one column to the other.
