DWA_01.3 Knowledge Check_DWA1

---

1. Why is it important to manage complexity in Software

It is important to manage complexity in software , because if you don't it can cause the software to crash which in a real life scenario could cause a company to collapse.
It also hampers group projects since bugs don't only affect your work on a project it affects everyone who is working on the project.
Also the more complex the project is the more things that can go wrong; making it crucial to keep your code under control or else it will lead to problems, cause confusion , and make it difficult to identify errors.

---

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

The sheer size of a project, the constant change to the code due to user demands, technical debt (forced to use complex code due to necessity and lack of time),functionality of the code (referring to what the code demands), and quality of code (referring to how clean ,maintainable and readable the code is
) are all factors of complexity.

---

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

We can manage complexity by:

-documenting (comments and JS DOCS)
-following Code Style Guide rules.,
-making the code modular (reusable),

- abstraction (breaking the code into smaller pieces)
These methods will make the code neater ,easier to understand , to reason about, and change when needed.

---

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

Yes there are implications of not managing complexity on a small scale. For instance, If you do not manage complexity on a small scale it will cause major problems if it were to be applied to bigger projects since it will become harder to find. It will also make it harder for someone else to understand, reason with , and edit your code .

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

-Use 2 spaces to indent: this enforces a consistent indentation making the code neater

- no unused vars: this eliminates unnecessary code (unused vars are vars that are only declared but are never used)

-infix operators must be spaced (const x = 2 not const x=2): it makes it easier to read and makes it easier to highlight errors.

-Keyword spaces (for example , if () not if()): this makes the neater and easier to read

---

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 in IWA 10. The IWA 10 section I was trying to fix checked for the earliest date of a set of dates. The logic was alright but it still

showed undefined on the console, however, after a long while I figured out that I used improper syntax (I used Date (xyz) instead of new Date (xyz)).
