DWA_01.3 Knowledge Check_DWA1

---

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

A big part is ensuring the maintainability of the software, as the software gets bigger and more complex we need to be able to maintain its complexity not only for ourselves but other people that may work on our code.

---

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

1. Poor Styling
2. Little / No Documentation
3. Creating too much technical Debt (quick fixes) and not going back to create cleaner code
4. No use of modular code or Abstraction

---

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

1. Creating Variable names that are Descriptive
2. Creating / updating Styling for better readability
3. Creating / updating documentation eg. JSDoc
4. Clearing technical Debt before it builds up
5. Creating / updating code to be used modular

---

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

Yes, You could come back to the code after a period of time and not understand what was written, wasting time / resources.

You can create / pickup bad coding habits

Quickly you can create complexity as your code grows introducing bugs as we are human at the end of the day and can get distracted.

---

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

Naming Conventions: An example would be naming a global Variable to use UPPER_SNAKE_CASE

const TOTAL_CHILDREN_PER_CLASS = 36

Code Indentation: An example would be when writing an object that we make it easier to see the nesting

```
const example {
        value: 50,
        name: Matt,
        Language: [english, afrikaans],
        Hobbies: {
                1: painting
                2: shooting
                3: gaming
        }
};
```

Line Length: Trying to keep code to 80 characters per line (don't think i need an example for this one)
Imports: Must not be line wrapped thus are exempt from the Line Length rule

---

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

I created a little flip card memory game
When i find two matching pairs i set the cards to show just a white card
However you can click on those white cards again and make them flip over when they

should of been disabled
I didn't really understand event listeners well and I tried to create if statements to fix it, however after lots of research and the final IWA project I realized there was a way to stop the event listener from firing once the two cards were selected.
