

# DWA\_01.3 Knowledge Check\_DWA1

---

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

To increase the functionality in Software, to avoid the creation of errors or bugs and to make the code more readable.

---

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

Using incorrect syntax, not using correct data types, using improper code structuring or using wrong variables and incorrectly naming and declaring variables in functions.

---

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

We can use code control measures such as proper variable naming, we can properly structure code by grouping related data together such as functions or arrays, and we can also add comments to label and differentiate between different sections of the code.

---

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

Bugs may not easily be found and easily fixable or avoidable.

---

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

Indentation: Use either tabs or spaces (usually 2 or 4 spaces) for indentation. Be consistent throughout your codebase.

Line Length: Keep lines of code reasonably short to enhance readability.

Variable Declarations: Prefer using `const` or `let` to declare variables and avoid using the outdated `var`. Use `const` for variables that won't be reassigned and `let` for variables that may be reassigned.

Naming Conventions: Use descriptive names for variables, functions, and classes. Generally, follow camelCase for variables and functions and PascalCase for classes and constructor functions.

Function Declarations: Use function declarations instead of function expressions for named functions whenever possible. For anonymous functions or function expressions, use arrow functions `() => {}` when they provide clearer syntax.

Object and Array Literals: Use consistent spacing and indentation when working with object literals and arrays. Generally, use a trailing comma after the last item in a multiline list for easier additions/removals.

Quoting Strings: Use single quotes `' '` or double quotes `" "` consistently for string literals. Choose one and stick to it throughout your codebase.

Comments: Include comments to explain complex logic, important decisions, or potential gotchas. Follow consistent commenting conventions within your codebase.

---

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

Braces and brackets when declaring functions and using functions that contain variable declarations as well as different event selectors and so forth. It took time to fix the bug because the functions and the whole structure of my code was too complicated to decipher.

---