# DWA_01.3 Knowledge Check_DWA1

---

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

Managing complexity in software is important for several reasons. As software systems grow in size and complexity, it becomes increasingly difficult to understand, maintain, and modify them. This can lead to increased development time, higher costs, and lower quality. By managing complexity, developers can create software that is easier to understand, maintain, and modify, which can result in faster development times, lower costs, and higher quality.

---

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

Size- As the size of a software system increases, so does its complexity. Larger systems have more components and interactions, making them more difficult to understand and manage.
Interdependence- When components of a software system are tightly coupled, changes to one component can have a ripple effect throughout the system, increasing its complexity.
Diversity- Software systems that support a wide range of functionality or serve a diverse user base can be more complex due to the need to accommodate different needs and requirements.
Technology-The use of new or rapidly changing technology can increase the complexity of a software system, as developers must constantly learn and adapt to new tools and techniques

---

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

Breaking code into smaller, more manageable modules can help reduce complexity and make it easier to understand and maintain.
Writing concise, readable code can help reduce complexity and make it easier to understand and maintain.

---

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

When complexity is not managed, it can lead to increased development time, higher costs, and lower quality. This is because as software systems grow in size and complexity, it becomes increasingly difficult to understand, maintain, and modify them. This can result in code that is difficult to understand and maintain, leading to increased development time and costs.

---

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

Variable names should be in lowerCamelCasing.
Add spaces around operators and after commas.
Always use two spaces for code indentation.
Keep JavaScript statements and objects on separate lines.
Keep JavaScript line length under 80 characters.
Use naming conventions.
Include code layout, whitespace, semicolon, and naming conventions.
Include node.js and JavaScript idioms
Include comments and documentation.

---

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

Javascript, When I  was doing the final project we were required to create  an online book store. It took me a few days to finish that project because I was struggling to fix the script file code so that the online book store would work perfectly.

---