DWA_01.3 Knowledge Check_DWA1

---

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

Excessive complexity increases an effort required for every feature delivery, reduces quality as an amount of bugs is also increased, and reduces functionality so that we can deliver in the same time frame.

---

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

Software engineering and software project management can be complex due to various factors, such as the dynamic nature of software development, changing requirements, technical challenges, team management, budget constraints, and timeline pressures.

---

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

By reducing the number of parameters, the number of nesting, and the number of statements in your function you can dramatically increase the readability and modularity of your code.

---

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

Because of the nonlinearity of the interactions, small causes can have large effects. The reverse is, of course, also true. The point is that the magnitude of the outcome is not only determined by the size of the cause, but also by the context and by the history of the system. This is another way of saying that we should be prepared for the unexpected. It also implies that we have to be very careful. Something we may think to be insignificant (a casual remark, a joke, a tone of voice) may change everything. Conversely, the grand five-year plan, the result of huge effort, may retrospectively turn out to be meaningless. This is not an argument against proper planning; we have to plan. The point is just that we cannot predict the outcome of a certain cause with absolute clarity.

---

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

-Consistently using the same style throughout your code makes it easier to read. Code that is easy to read is easier to understand by you as well as by potential collaborators. Therefore, adhering to a coding style reduces the risk of mistakes and makes it easier to work together on software.
-Programming style, also known as code style, is a set of rules or guidelines used when writing the source code for a computer program. It is often claimed that following a particular programming style will help programmers read and understand source code conforming to the style, and help to avoid introducing errors.

---

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

A day or half day works well on average, especially since most bugs are coding bugs (on average 3 hours) or data bugs (6.5 hours). Even design bugs on average only take little more than a day to resolve.
