DWA_07.4 Knowledge Check_DWA7

1. Which were the three best abstractions, and why?

Using Closures:

- Closures are functions that retain access to variables from their outer lexical environment even after the outer function has returned.

2. Which were the three worst abstractions, and why?

Using Classes (ES6):

- ES6 introduced the `class` syntax, allowing you to define classes and objects more structurally.

3. How can The three worst abstractions be improved via SOLID principles.

Single-Responsibility Principle (SRP)

Ensure that each abstraction (class, module, etc.) has only one reason to change

Open-Closed Principle (OCP

Design abstractions to be open for extension (new features) but closed for modification (existing code remains untouched)

Liskov Substitution Principle (LSP)
Solution: Derived classes should be substitutable for their base classes without altering program correctness