**Q4**

One instance where using a design pattern may not be beneficial is when it introduces unnecessary complexity to a simple problem. This was a key point made during our Introduction to Design Pattern class discussions. Design patterns are powerful tools meant to solve common software design challenges, but if applied blindly or in situations where a straightforward solution would suffice, they can make the codebase more convoluted.

For example, using the Abstract Factory or Decorator pattern in a small-scale project with minimal variation or no need for dynamic behavior can be overkill. Instead of aiding clarity, it can actually confuse collaborators and slow down development, especially for teams unfamiliar with that pattern’s structure.

So, while patterns are meant to promote reusability and scalability, applying them without justification can violate the KISS (Keep It Simple, Stupid) principle.

Sources for the example

<https://softwareengineering.stackexchange.com/questions/49379/when-should-i-use-and-not-use-design-patterns>

<https://www.geeksforgeeks.org/abstract-factory-pattern/>