

Design Patterns

Established conventions to solve a software problem.

- promote reusability of code and lower maintenance
- easier for you and others to read/understand
- **Scalability and Reusability**

Why?

- they increase coding speed
- they bring structure to our code
- boost efficiency

Singleton Design Pattern

- A singleton is a design pattern to not allow more than one instance of a class to exist.
- Generally in Java, we achieve this by using a private constructor
- **Lazy**: will wait until an instance is needed to create one
- **Eager**: will create an instance right away (at start of application)

Factory Design Pattern

- a solution to write code that creates objects whose type will not be known at runtime
- when we use a factory, we know what we want an object to be able to do, but we don't care what the final product looks like