



AFS

Agile Full Stack
Developer Bootcamp
Thoughtworks @

Design Pattern

/thoughtworks



AFS

Agile Full Stack
Developer Bootcamp
Thoughtworks @

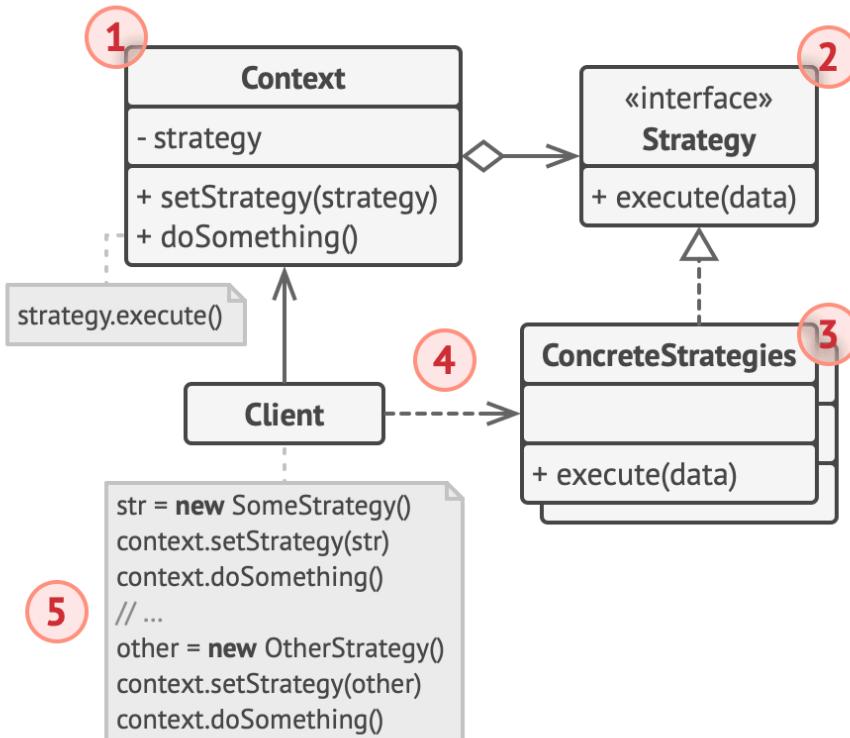
Strategy Pattern

/thoughtworks

Strategy Pattern

Strategy is a behavioral design pattern that lets you define a family of algorithms, put each of them into a separate class, and make their objects **interchangeable**.

Strategy Pattern



1. The **Context** maintains a reference to one of the concrete strategies and communicates with this object only via the strategy interface.
2. The **Strategy** interface is common to all concrete strategies. It declares a method the context uses to execute a strategy.
3. **Concrete Strategies** implement different variations of an algorithm the context uses.
4. The context calls the execution method on the linked strategy object each time it needs to run the algorithm. The context doesn't know what type of strategy it works with or how the algorithm is executed.
5. The **Client** creates a specific strategy object and passes it to the context. The context exposes a setter which lets clients replace the strategy associated with the context at runtime.



AFS

Agile Full Stack
Developer Bootcamp
Thoughtworks @

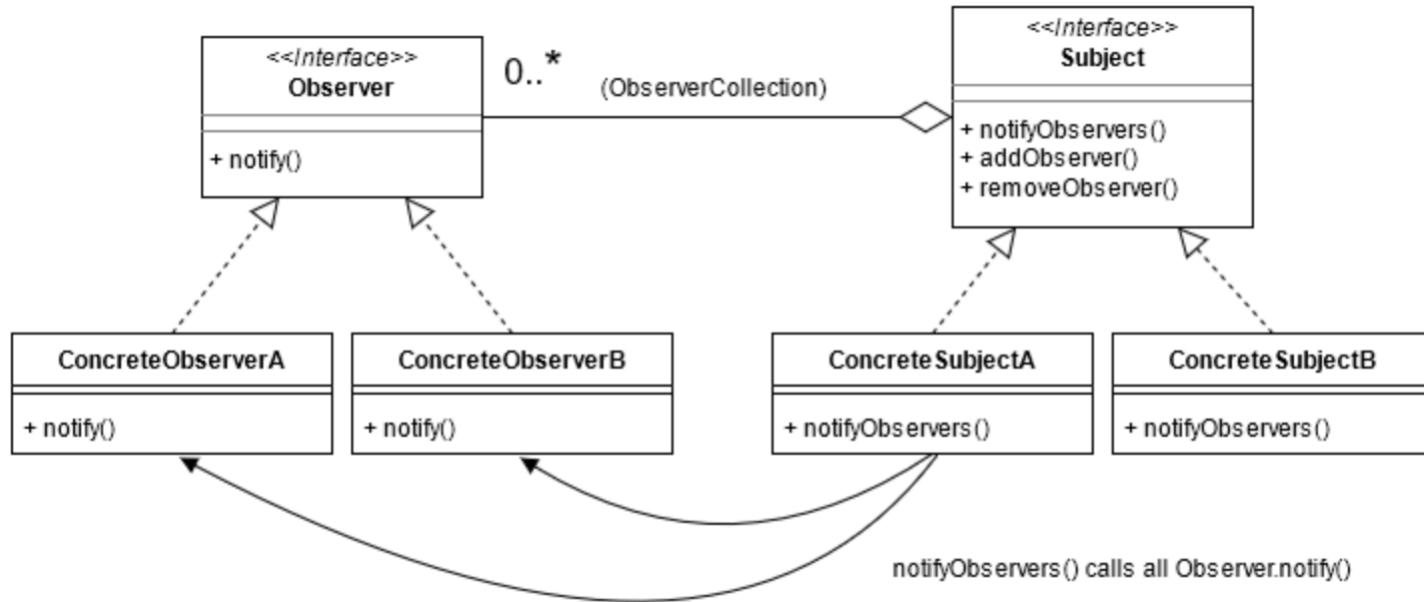
Observer Pattern

/thoughtworks

Observer Pattern

Observer pattern used when there is a one-to-many relationship between objects. For example, when an object is modified, it will automatically notify the dependent objects. The observer pattern is a behavioral pattern.

Observer Pattern





AFS

Agile Full Stack
Developer Bootcamp

Thoughtworks ©

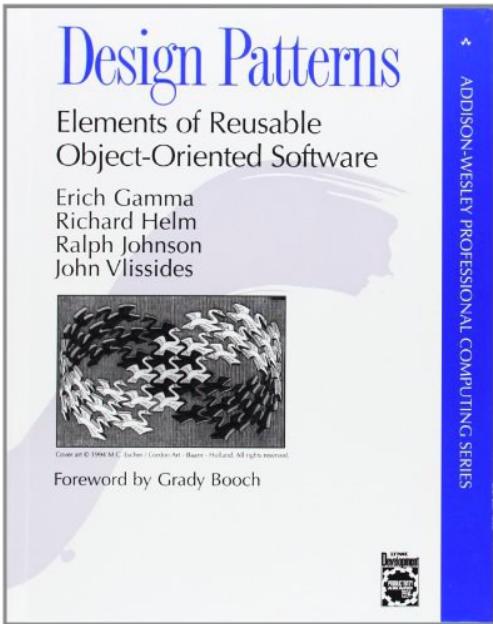
Design Pattern Presentation Summary

/thoughtworks

Suggestions on Design Pattern

1. First **understand** the design pattern and **try to use** it
2. **Don't get obsessed** after the first step. It is better for you to understand the **design principles** behind the design pattern.

Books and references



<https://refactoring.guru/design-patterns>



AFS

Agile Full Stack
Developer Bootcamp
Thoughtworks @

Thank You !

/thoughtworks