SDET Course

Design Patterns - Introduction

Why??

- Code readability and efficiency
- Improves communication
- Improves code quality
- Good software design practices

3 Types of Design Patterns

- **Creational -** These patterns are all about class instantiation or object creation. They can be further divided into class-creation patterns and object-creational patterns.

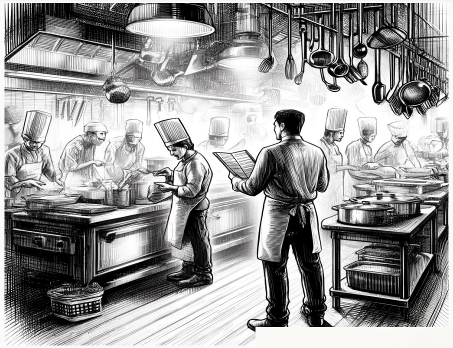
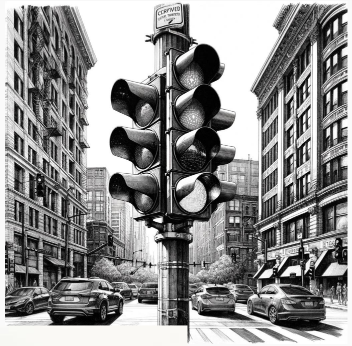
- **Structural -** These patterns deal with object composition or the structure of classes. They help ensure that if one part of a system changes, the entire system doesn't need to do the same.

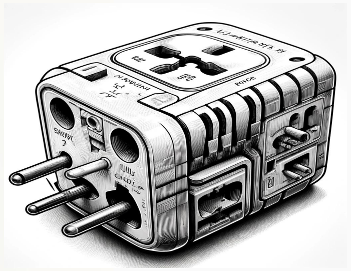
- **Behavioral -** These patterns are all about class's objects communication. They help define how objects interact in a way to increase flexibility in carrying out communication.

3 Types of Design Patterns

- Creational
    - Singleton
    - Builder
    - Prototype
    - Factory Method
    - Abstract Factory
- Structural
    - Adapter
    - Composite
    - Proxy
    - Flyweight
    - Bridge
    - Facade
    - Decorator
- Behavioral
    - Strategy
    - Observer
    - Command
    - Memento
    - State
    - Template Method
    - Mediator
    - Chain of Responsibility
    - Interpreter
    - Visitor
    - Iterator

Some Examples From Reality



How to Learn?

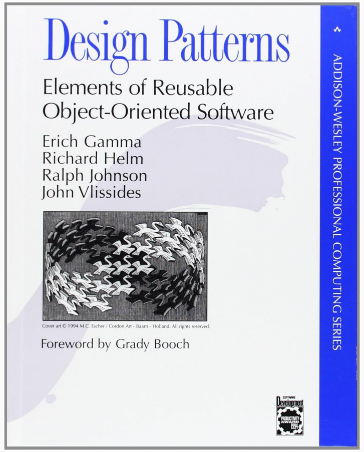
- Create a personal repository
- Follow the video
- Try it yourself
- Don’t forget to document
- Write short summary in code to remember

Sources

Head First Design Patterns: Building Extensible and Maintainable Object-Oriented Software 2nd Edition.
Buy it on Amazon

O'REILLY
Second Edition
Head First
Design Patterns
Building Extensible & Maintainable Object-Oriented Software
Eric Freeman & Elisabeth Robson
with Kathy Sierra & Bert Bates
A Brain-Friendly Guide

Design Patterns: Elements of Reusable Object-Oriented Software.
Buy in on Amazon


Design Patterns
Elements of Reusable Object-Oriented Software
Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides
Foreword by Grady Booch
ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES

Learning Steps

- Description
- General use case
- Diagram
- Code sample (Java)
- Use case in test automation

My Secret Recipe

How I created each presentation WITH working code example in less than 15 minutes

How each chapter of this course was created:

1. Make a copy of the previous presentation and change the name to the next design pattern
2. Find the appropriate wikipedia page and copy the diagram
3. Use the following prompts in ChatGPT:
    a. write a short paragraph about what is the XXX design pattern
    b. list 6 example when this design pattern is useful
    c. list some examples from the software test automation area
    d. give an example of how this pattern can be useful for the XXXX use case [when writing selenium test automation scripts]
    e. write a java code [selenium] example which demonstrate this use case

So, let’s get to work...