Design Patterns

Contents

Overview

Design Patterns represent best practices developed and used by experienced software developers.

Design patterns can generally be found with object-oriented languages. In general, they are solutions to common problems that developers faced during development. Think of them as "templates" that you can use to make your code more efficient or effective.

There are currently 23 well-known Design patterns, which are all from this 1994 textbook.

The usage of design patterns focus in two areas:

- **Standardisation** – design patterns offer a standard of terms in programming.
- **Best practice** – tested methods of programming that have been refined throughout the years.

Design patterns can be classified into 3 categories:

- Creational – focuses on creating objects, whilst hiding the logic that creates them.
- Structural – focuses on the structure of classes and objects, and emphasises the use of inheritance and interfaces to define the composition of objects.
- Behavioural – focused on the communication between objects.

Tutorial

Not applicable.

Exercises

Not applicable.

- Javascript
- Spring Boot
- Selenium
- Sonarqube
- Advanced Testing (Theory)
- Cucumber
- MongoDB
- Express
- NodeJS
- React
- Express-Testing
- Networking
- Security
- Cloud Fundamentals
- AWS Foundations
- AWS Intermediate
- Linux
- DevOps
- Jenkins Introduction
- Jenkins Pipeline
- Markdown
- IDE Cheatsheet