Table of Contents

Software design pattern

Reusable solution to a commonly occurring problem within a given context in software design.

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

*Snippet from Wikipedia: **Software design pattern***

> In software engineering, a **software design pattern** is a general, reusable solution to a commonly occurring problem within a given context in software design. It is not a finished design that can be transformed directly into source or machine code. Rather, it is a description or template for how to solve a problem that can be used in many different situations. Design patterns are formalized best practices that the programmer can use to solve common problems when designing an application or system.
>
> Object-oriented design patterns typically show relationships and interactions between classes or objects, without specifying the final application classes or objects that are involved. Patterns that imply mutable state may be unsuited for functional programming languages. Some patterns can be rendered unnecessary in languages that have built-in support for solving the problem they are trying to solve, and object-oriented patterns are not necessarily suitable for non-object-oriented languages.
>
> Design patterns may be viewed as a structured approach to computer programming intermediate between the levels of a programming paradigm and a concrete algorithm.

Creative Commons Attribution-Share Alike 4.0

Related:

- Composable Architecture

External links:

- https://sourcemaking.com/design_patterns
- https://refactoring.guru/design-patterns/catalog
- https://www.tutorialspoint.com/design_pattern/design_pattern_overview.htm
- https://www.oodesign.com/
- https://dzone.com/articles/what-is-design-pattern
- https://nikku1234.github.io/2020-09-18-Software-Design/
- 10 Common Software Architectural Patterns in a nutshell — *towardsdatascience.com*
    1. Layered pattern
    2. Client-server pattern
    3. Master-slave pattern
    4. Pipe-filter pattern
    5. Broker pattern
    6. Peer-to-peer pattern
    7. Event-bus pattern
    8. Model-view-controller pattern
    9. Blackboard pattern
    10. Interpreter pattern

kb, architecture, programming, devopscreate, devopsverify, taxonomy, pattern

From:
https://almbok.com/ - **ALMBoK.com**

Permanent link:
https://almbok.com/kb/software_design_pattern

Last update: **2023/01/21 12:30**