Design Patterns

Ajay Panchal
ajay.panchal@zensar.com

What is Design Pattern ?

- A design pattern is proved solution for solving the **specific problem/task.**
- Design patterns are solutions to general problems that developers faced during software development.
- So Let's take one **Problem Statement:** Suppose you want to create a class for which only a single instance (or object) should be created and that single object can be used by all other classes.

Types of Design Pattern.

- Creational Design Pattern
- Structural Design Pattern
- Behavioral Design Pattern

Creational Design Pattern

These design patterns are all about **object creation**

- Factory Pattern
- Abstract Factory Pattern
- Singleton Pattern
- Prototype Pattern
- Builder Pattern

Singleton Pattern

- **Definition:**

The singleton pattern is a design pattern that **restricts the instantiation** of a class to **one object.**

- **Static member:** It gets memory only once because of static, it contains the instance of the Singleton class.
- **Private constructor:** It will prevent to instantiate the Singleton class from outside the class.
- **Static factory method:** This provides the global point of access to the Singleton object and returns the instance to the caller.

Structural Design Pattern

Structural design patterns are concerned with how classes and objects can be composed, to form larger structures.

- Adapter Pattern
- Bridge Pattern
- Composite Pattern
- Decorator Pattern
- Facade Pattern
- Flyweight Pattern
- Proxy Pattern

Adapter Pattern

Definition:

- The adapter pattern **convert the interface of a class into another interface clients expect**.
- The Adapter Pattern is also known as **Wrapper.**

Behavioral Design Pattern

Behavioral patterns are about identifying common communication patterns between objects and realizing these patterns.

- Chain Of Responsibility Pattern
- Command Pattern
- Interpreter Pattern
- Iterator Pattern
- Mediator Pattern
- Memento Pattern
- Observer Pattern

Observer Pattern

- **An Observer Pattern** says that "just define a one-to-one dependency so that when one object changes state, all its dependents are notified and updated automatically
- The observer pattern is also known as Dependents or Publish-Subscribe.

Advantages of Design Pattern

- They are reusable in multiple projects.
- They provide the solutions that help to define the system architecture.
- They capture the software engineering experiences.
- They provide transparency to the design of an application.
- They are well-proved and testified solutions since they have been built upon the knowledge and experience of expert software developers.
- Design patterns don't guarantee an absolute solution to a problem. They provide clarity to the system architecture and the possibility of building a better system.

Thank You