



Workshop on Design patterns - 3 Days (24 hours)

Training Style

Purpose of this training is not just to explain design patterns and their usage, but most importantly to bring design pattern in day to day talk of the participants.

Some practices we follow to ensure that participant remember each and every pattern is as follows

1. At the very start, each participant is named as a design pattern and participant is called by that name only.
2. During the course, participants need to be in random groups and share their understanding, confusions and queries so far.
3. All sessions are kept as interactive and practical as possible.
4. The complete training program is converted into a Question and Answer with very less theoretical talk.
5. Participants are supposed to fill the feedback form for every session and not at the end of the course.
6. Every day ends with a quick quiz and recap session.

Training Objectives

At the end of the training, participants will be able to

1. Immediately start applying the patterns learnt to solve new design problem statements and improve existing designs.
2. Distinguish design patterns from architectural patterns and idioms
3. Distill existing and new design solutions found by them in their work and document such solutions to share the knowledge or for future use.
4. Define the value of applying design patterns to fellow software engineers and inspire them to adopt.
5. Define and understand the meaning and purpose of Object Oriented Programming (OOP) fundamentals with sound clarity and apply them correctly.

Target Groups:

1. Experienced and fresh software developers who want to get a firm grasp of design patterns.
2. New job seekers with software related academics/programming background interested in gaining professional knowledge of software systems designing (Design pattern knowledge is surely to add marks in job interviews).
3. Application Architects.

Prerequisites:

1. Basic familiarity with an object oriented programming language (Like Java, C++, C# or PHP with OOP) is essentially required. But you may or may not know about how OOP concepts used in practical world.

Training Content:**1. Introduction**

- What are design patterns?
- Why patterns are important?
- Why do we need design pattern?
- Where to apply design patterns and where to ignore them?

2. Essentials/Concepts

- Elements of a pattern
- Pattern documentation
- Anti-patterns

3. Pattern Classification

- Pattern systems and pattern languages
- POSA classification scheme
- Gang-of-Four classification scheme

4. POSA Patterns

- Architectural patterns
- Design patterns
- Idioms

5. OOP Fundamentals (Pragmatic Nature And Clearing Confusions)

- Abstraction
- Encapsulation
- Inheritance
- Polymorphism

6. Gang-of-Four Design Patterns

- Creational patterns
- Structural patterns
- Behavioral patterns

7. Using design patterns during

- Development
- Defect fixing and
- Refactoring.

8. Why it is perceived that using Design Patterns is complicated and How to make it simple?**9. Design Pattern games, quiz and winners****10. Training Recap (every day)**

Design patterns which will be covered during the training sessions**Creational patterns**

- Abstract factory
- Builder
- Factory method
- Lazy initialization
- Object pool
- Prototype
- Singleton
- & More

Structural patterns

- Adapter/Wrapper
- Bridge
- Composite
- Decorator
- Facade
- Flyweight
- Proxy
- & More

Behavioral patterns

- Blackboard Generalized observer
- Chain of responsibility
- Command
- Interpreter
- Iterator
- Mediator
- Memento
- Observer
- Servant
- State
- Strategy
- Template
- Visitor
- & More