JAVA DESIGN PATTERNS (4 Days)

Day 1:

- Basic Class Diagram Notations
- Delegation vs. Inheritance
- Single Responsibility Principle
- Open closed principle
- Liscov Substitution Principle
- Interface Segregation Principle
- Dependency Inversion Principle
- Once and only once principle or DRY principle
- Samurai Principle
- Fundamentals of Clean coding
- Introduction to Design Patterns.
- For the following design patterns give an explanation, class diagram, and examples. If necessary then discuss the code.
- Strategy
- Observer

Day 2:

For the following design patterns give an explanation, class diagram, and examples. If necessary then discuss the code.

- Decorator
- Factory and Abstract Factory
- Singleton
- Command
- Adapter
- Facade

Day 3:

For the following design patterns give an explanation, class diagram, and examples. If necessary then discuss the code.

- Template
- Iterator
- Composite
- State
- Proxy. Use CircuitBreaker to make the code secure.
- Bridge
- Builder

Day 4:

For the following design patterns give an explanation, class diagram, and examples. If necessary then discuss the code.

- Flyweight. Using flyweight to make the code more secure.
- Interpretor
- Mediator
- Memento
- Prototype
- Visitor

Exercises and Case study on Design Patterns.

Note:

- Topics may get shifted from one day to another based on participants' feedback. All topics will get covered by the end of the course.
- All participants must know Java SE.