


2024 - Java / Design patterns + TDD / QA

Document status	
Document owner	Pankaj Wadhwa Sudheer Parauha Ravindra Ghodasara Soumitra Soundankar Aditya-AmrutDeshpande
Delivered by	Vinsys
Theme	Build Architecture
Sub theme	Local development Microservicesarchitecture
Module	Java / Design patterns TDD / QA
Duration	4 days

Pre-requisites

N/A

Learning Objectives

Upon completion of this module, the participant will be able to:

- Understand basics of Java coding, debugging and code optimization Understand the Java coding best practices
- Understand SOLID principles
- Understand some of the key creational, structural and behavioral patterns as defined by GoF book Understand how the patterns are implemented in core Java libraries
- Expanding from Microservices to APIs
- Gain an understanding of the Microservices architecture and pros/cons of this architecture approach Gain an understanding of Monolith versus Microservices versus Serverless architecture styles Understand common design patterns used in Microservices development
- Understand some of the popular tools and frameworks in use for Microservices development
- High level approach to Testing and Software Quality Rationale of developer writing test cases
- TDD: Process and its pitfall
- Basic understanding of other practices like BDD and how it differed from TDD Testing Framework (Mockito, Wiremock, etc.)
- What is QA ?
- What is Software Testing? Definition, Basics & Types Basic of MTS (Minimum Testing Standards)

Day 1

- Java evolution
- where is Java good to use where not to use Java
- Core libraries (io, database, collections)
- Debugging
- Functional programming
- Java 8 Features- Lambdas and Streams

Day 2

- Concurrency
- Program Documentation
- SOLID principles
- GoF patterns - Some common Creational, Structural, Behavioral Design Principles
 - *Singleton, Factory, Builder, Bridge, Decorator, Proxy, Command, Template and Strategy*
- Java Do's and Don'ts/ best practices -> checklist is at <https://mydb.intranet.db.com/docs/DOC-514588>
- **Introduction to Java reflection and annotation processing**

Day 3

- Software quality and basic of Unit testing
- Definition of Unit and technique to test Unit in isolation. Example of testing simple math like units
- Introduce JUnit framework and basic hands on examples. BDD Orientation
- Introduction to Sonar
- Introduction to Spring framework
- Getting started with Spring Boot, Creating REST application with 3 tier architecture
- **Spring boot Actuator, Key challenges Monitoring/Alerting**

Day 4

- Introduction to Microservices
- Understanding the Monolith application and its challenges What are Microservices
- Basic building blocks of Microservices
- How do Microservices solve the challenges with Monolith?
- Microservices in action
- How to build, deploy and test Microservices Common design patterns for creating Microservices REST and JSON
- Popular frameworks in use Key benefits
- Introduction to Spring cloud
- **Messaging with rabbitMQ/ActiveMQ**

Pre-reading, Resources, Hands-on sessions / exercises

2024 - Java / Design patterns + TDD / QA

Execute Full GC for JVM instance via JMX

How to update Java on DAP instance tc0322

How to update Java on DAP instance tc0235

How to update Java on SMMP

Assignments and Evaluations (optional)

N/A