

Application Development Upskilling Program

CONFIDENTIAL

Table of Contents

Application Development Upskilling Program - Data Structures, Java 11 Features, Spring Boot & Design Patterns	4
1. Course Outline	4
1.1. Data structures and Algorithms	4
1.2. Java 11 Features.....	4
1.3. Spring Boot with Microservices	4
1.4. Design Patterns	4
2. Pre-requisites	5
3. Program Schedule	5
4. Program Details	5
1. UML (1 Hr)	5
1.1. Introduction	5
1.2. Modelling Concepts	5
1.3. Class Modelling	5
1.4. State Modelling.....	6
1.5. Interaction Modelling	6
2. Data Structure and Algorithms (6 Hrs)	6
2.1. Core Data Structures.....	6
2.2. Core Collection Classes	6
2.3. Collection Sorting.....	6
2.4. Working with Generics.....	6
3. Java 11 Features – 8 hours	6
4. Design Patterns – Java 11- 8 hours	7
4.1. Introduction to Design Patterns.....	7
4.1.1. Object Orientation Basics.....	7
4.1.2. Object Orientation principles.....	7
4.1.3. Advantages of Design Patterns	7
4.2. Reviewing Object-Oriented Principles in Java	7
4.3. Reviewing Gang of Four Patterns.....	7
5. Spring Boot with Microservices- 14 hours.....	8
5.1. Microservices and RESTful APIs with Spring Boot and Spring Cloud (1 hr)	8
5.2. Restful Web Services with Spring Boot (9 hrs).....	8
5.3. Unit Testing & Mockito (2hrs).....	9
5.4 DEVOPS (2 hrs)	9

6. Microservice - Design Patterns (2 Hrs – Mostly theory)	9
6.1. Decomposition Patterns	9
6.2. Integration Patterns.....	9
6.3. Database Patterns.....	9
6.4. Observability Patterns.....	10
6.5. Cross-Cutting Concern Pattern	10

CONFIDENTIAL

Application Development Upskilling Program - Data Structures, Java 11 Features, Spring Boot & Design Patterns

1. Course Outline

1.1. Data structures and Algorithms

By the end of this module participants shall have basic and intermedia concepts of Data Structure. Data Structure make use to store and organize data so that it can be used efficiently.

1.2. Java 11 Features

Java is the world's most popular programming language. It is easier to use, write, compile, debug, and learn than any other programming language.

In this course, you will go through real-world examples and work with Java 11 programming concepts and features. You will bridge the gap between learning and perform and improve on your development skills.

By the end of the course, you will have a firm grasp of using Java 11 in your day-today programming. This course will be the first step in your career to becoming a professional Java developer.

1.3. Spring Boot with Microservices

Microservices is an architectural style in software development life cycle based on SOA (Service Oriented Architecture) to manage and fine grain a giant application. Spring Boot is used to integrate and deploy microservices

By the end of this course, participants would be able to write a restful service, and create a micro service architecture with OSS tools like Zuul, Eureka, Ribbon, Feign etc.

1.4. Design Patterns

In software engineering, a design pattern is a general repeatable solution to a commonly occurring problem in software design. A design pattern isn't a finished design that can be transformed directly into code. It is a description or template for how to solve a problem that can be used in many different situations.

By the end of this course, participants will know new features of Java 11 and improvements of the JVM. And have the maturity to write a clean code which is more collaborative and appreciate the concept of WORM – Write Once Read Many and develop Highly Cohesive, Loosely Coupled system.

2. Pre-requisites

- Good understanding of Java programming language
- Working with Web application will be an advantage.

3. Program Schedule

Training Day	Duration in Hours	Module
Day 1	1	UML
	3	Introduction to Data Structure
Day 2	3	Data Structures & Algorithms
Day 3	4	Java 11 features Part 1
Day 4	4	Java 11 features Part 2
Day 5	4	Introduction to Design Patterns
Day 6	4	Working with Design Patterns
Day 7	4	Introduction to Spring Boot
Day 8	4	Spring Boot with Microservices + Netflix OSS Part 1
Day 9	4	Spring Boot with Microservices + Netflix OSS + Unit Testing Part 2
Day 10	4	Devops + Microservice Design Patterns

4. Program Details

1. UML (1 Hr)

1.1. Introduction

- About Object Orientated Technology, Development and OO Modelling History.

1.2. Modelling Concepts

- Modelling design Technique, Three models, Class Model, State model and Interaction model.

1.3. Class Modelling

- Object and class concepts, link and association, Generalization and Inheritance, Advanced class modelling- aggregation, Abstract class metadata, constraints.

1.4. State Modelling

- Event, state, Transition and conditions, state diagram, state diagram behaviour, concurrency, Relation of Class and State models.

1.5. Interaction Modelling

- Use case Models, sequence models, activity models

2. Data Structure and Algorithms (6 Hrs)

2.1. Core Data Structures

Searching and Hashing algorithms. Search algorithms – Sequential Search, Ordered lists, binary search. Searching using Hashing. Hash tables. Various types of sorting

2.2. Core Collection Classes

- The Collections Framework
- The Set Interface
- Set Implementation Classes
- The List Interface
- List Implementation Classes
- The Queue Interface
- Queue Implementation Classes
- The Map Interface
- Map Implementation Classes

2.3. Collection Sorting

- Using Java Features with Collections
- Sorting with Comparable
- Sorting with Comparator
- Sorting Lists and Arrays
- Collections Utility Methods
- Tuning ArrayList
- Tuning HashMap and HashSet

2.4. Working with Generics

- Why Use Generics?
- Java Generics using Map
- Generic class
- Type Parameters
- Generic Method
- Wildcard in Java Generics (?, extends, super)

3. Java 11 Features – 8 hours

- Setting up Environment and Prerequisites
- Running Hello World
- Writing your first Java 11 program
- Creating the basic structure, basic class
- Using primitive types
- Exploring strings

- Local variable syntax for Lambda Parameters and low Overhead GC10.
- Hide Your Data – Encapsulation
- Hide Your Implementation with Abstraction
- Inheritance
- Polymorphism
- Documentation using Javadoc
- Working with Strings, Characters and Regular Expressions
- Error handling in Java 11
- Managing Input/output
- Using Generics and Collections
- Exploring Lambdas and Streams
- Working with Annotations
- Using the Date, Time and Money API
- Working with Internationalization
- Launch single-file-source-code programs
- Security updates
- Java virtual machine (JVM) improvements
- Java 11 deprecations and housekeeping

4. Design Patterns – Java 11- 8 hours

4.1. Introduction to Design Patterns

4.1.1.Object Orientation Basics

- Abstraction
- Encapsulation
- Polymorphism
- Inheritance

4.1.2.Object Orientation principles

- Encapsulate what varies
- Favor composition over inheritance
- Program to interfaces, not implementation

4.1.3.Advantages of Design Patterns

4.2. Reviewing Object-Oriented Principles in Java

- Describe how OO concepts apply to Java
- Describe how OO principles apply to Java
- List the goals of an OO language
- Interpret Unified Modelling Language (UML) notation and create UML diagrams
- Identify selected design patterns

4.3. Reviewing Gang of Four Patterns

- List key behavioural, creational and structural patterns
- Apply the Facade pattern
- Apply the Strategy pattern
- Apply the Composite pattern
- Review the Model-View-Controller (MVC) patterns

5. Spring Boot with Microservices- 14 hours

5.1. Microservices and RESTful APIs with Spring Boot and Spring Cloud (1 hr)

- Introduction to Web Services
- What is a Web Service?
- Important How Questions related to Web Services
- Web Services - Key Terminology
- Introduction to SOAP Web Services
- Introduction to RESTful Web Services
- SOAP vs RESTful Web Services

5.2. Restful Web Services with Spring Boot (9 hrs)

- Introduction - RESTful Web Services with Spring Boot
- Initializing a RESTful Services Project with Spring Boot
- Understanding the RESTful Services
- Creating a Hello World Service
- Enhancing the Hello World Service to return a Bean
- Quick Review of Spring Boot Auto Configuration and Dispatcher Servlet
- Implementing GET Methods for User Resource
- Implementing POST Method to create User Resource
- Enhancing POST Method to return correct HTTP Status Code and Location
- Implementing Exception Handling - 404 Resource Not Found
- Implementing Generic Exception Handling for all Resources
- Implementing DELETE Method to delete a User Resource
- Implementing Validations for RESTful Services
- Implementing HATEOAS for RESTful Services
- Overview of Advanced RESTful Service Features
- Internationalization for RESTful Services
- Versioning RESTful Services - Basic Approach with URIs
- Implementing Basic Authentication with Spring Security
- Overview of Connecting RESTful Service to JPA
- Creating User Entity and some test data
- Updating GET methods on User Resource to use JPA
- Updating POST and DELETE methods on User Resource to use JPA
- Implementing a GET service to retrieve all Posts of a User
- Implementing a POST service to create a Post for a User
- Richardson Maturity Model
- Understanding the need of API Gateway
 - Differentiating ZUUL and Spring Cloud Gateway
 - Implementation Spring Cloud Gateway
 - Introduction to circuit breaking and implementing with Hystrix
 - Documenting the service end points create with swagger
 - Introduction to Sync/Async way of programming
 - Securing microservice
 - Implementing JWT with Spring Cloud Gateway

- Monitoring the services with Actuator and HAL browser

5.3. Unit Testing & Mockito (2hrs)

- Introduction to Unit testing
- Understanding JUnit 4 framework
- Working with @Test, @Before, @After etc
- Working with exception handling
- Introduction to Mockito
- Setting up Mockito
- Working with annotation based mock, @Mock, @InjectMock, @MockBean
- Verifying mock
- Working with Mockito spy
- Writing Unit test cases for the services
- Mocking the services such that DB values are not changed
 - Validating the code coverage and best practices of the code written
 - Integrating the spring boot application with SonarQube

5.4 DEVOPS (2 hrs)

- Understanding containerization and virtualization
- Introduction to Docker
- Writing commands with Docker pull
- Understanding docker HUB
- Understanding docker images and docker process and managing the life cycle
- Deploying SpringBoot application with Docker
- Seeding simple springboot application on GITHUB
- Introduction to Jenkins – understanding simple Jenkins project, pipeline project etc
- Deploying SpringBoot application with Jenkins

6. Microservice - Design Patterns (2 Hrs – Mostly theory)

6.1. Decomposition Patterns

- Decompose by Business Capability
- Decompose by Subdomain
- Strangler Pattern

6.2. Integration Patterns

- API Gateway pattern
- Aggregator pattern
- Client Side UI Composition Pattern

6.3. Database Patterns

- Database per service
- Shared Database Per Service
- Command Query Responsibility Segregation (CQRS)
- Saga Pattern

6.4. Observability Patterns

- Log Aggregation
- Performance Metrics
- Distributed Tracking

6.5. Cross-Cutting Concern Pattern

- External Configuration
- Service Discovery Pattern
- Circuit Breaker Tracking
- Blue Green Deployment Pattern

CONFIDENTIAL