|  |
| --- |
| **Duration : 8 days** |

**Java version used jdk 17**

**Spring boot version 3.x**

[**https://docs.oracle.com/javase/tutorial/**](https://docs.oracle.com/javase/tutorial/) **from this cover**

1. Learning java language
2. Essential java classes
3. Collections

**Session Plan**

Approach

1. Convert the existing core java application into spring boot application without http, use command line runner to check working of the core java application.
2. Learn the jpa implementation without using any web or service by using command line runner interface.
3. Convert the dao layer into jpa in the earlier assignment.
4. Figure out http related request processing and response generation without dealing with service. Identity security needs and also add security layer to the web service.
5. Once http is done, add a REST based web services layer on the service layer built.
6. Do unit testing on the application built and ensure Jenkins pipe line is introduced and intro to how to deploy an spring boot application on AWS

**Day 1 and Day 2** (two days so that any basics missing in spring like life cycle can be covered.

1. Introduction to 3 tier application, role of presentation, service and dao.
2. Event Handling (this will help in understanding command line runner and http request handling)
   1. Things done by Event source.
   2. Things done by Event listener.
3. Concept of build tool using Graddle
4. Graddle basics.
   1. Build run test and configuration settings.
5. Introduction to Spring boot
   1. Basics
      1. Creation of Object and wiring.
   2. Understanding how annotations work
      1. @SpringBootApplication
      2. @Component
      3. @Service
      4. @Repository
      5. @ComponentScan
   3. Concept of autoconfiguring.
   4. Concept of starters.
   5. Configuration using properties files.
   6. Profiles.
   7. Know
      1. Architecture
      2. Which objects spring has got with it.
   8. Command Line runner or AppicationRunner
      1. Check whether the 3 tier application works written in core java works in spring boot.
   9. Convert the core java 3 tier application into spring boot application.

**Day 3**

1. Introduction to JPA and implementation using Hibernate.
   1. Quickly talk about JDBC based approach and then talk about JPA.
   2. Properties to use in the configuration.
   3. Concept of Entity classes
   4. Modelling Entity classes and tables.
   5. Doing crud operation in JPA
   6. Prove the crud operation of the JPA works , by using command line runner.
   7. Convert the 3-tier application which was using Datastructure into JPA.

**Day 4**

1. Introduction to Web
   1. Category of web usage
      1. Web for presentation
      2. Web as a wrapper on the service layer.
      3. Role of JSON and XML in web services to represent input and output.
      4. Distributed computing basics.
      5. How to process request
         1. Read parameters via request param
         2. Read parameters via path variable
         3. Read parameters via requestBody
         4. Read http header information
         5. Mode of request.
      6. Response generating
         1. Model generation
         2. http status codes usage
         3. HTTP Entity usage.
      7. Learn to work on http without touching database or service.
      8. Exception handling and validation in web tier.
      9. Debugging distributed applications.
   2. Integrate
      1. Web service with actual service layer which uses DAO using JPA.

**Day 5**

1. Calling web services
   1. RestTemplate/RestClient
   2. WebClient
2. Introduction to Securing web services.
   1. Basic Http authentication
   2. JWT
   3. Oauth
3. Unit testing and Mocking
   1. Introduction to JUnit
   2. Unit testing of Spring boot application
   3. Mocking basics using Mockito
   4. Introduction to Cucumber.
   5. Testing rest applications with Karate

**Day 6**

1. Introduction to Microservices architecture
2. Introduction to API Gateway Eureka
3. Microservices introduction
4. Spring Cloud introduction and its modules.
5. Deploying application using Eureka and calling services by using Feign.
6. Some commonly used patterns in Microservices.
7. Meaning of Circuit Breaker, Rate Limiter, Retry.
8. Using Resilience4J
9. Modules available in Resilience4j
10. Circuit breaker states, applying the Rate Limit, Bulkhead ,Cache. And Time Limiter.

Aspect Oriented programming

1. Introduction to Aspect Oriented programming
2. Aspect, Advice and Pointcut Proxy and weaving
3. Some commonly used annotations in Aspect oriented programming example security and logging.
4. Types of Advice
5. Custom example Aspect oriented programming to demonstrate.
6. Build a custom annotation and use aspect-oriented programming to tell how long a resource was acquired and when it was released.

**Day 7**

1. a. Jenkins introduction
   * 1. Job/project
     2. Build
     3. Pipeline
     4. Node
     5. Workspace
   1. Understanding the containerization, role of Docker, Kubernetes.
   2. Making docker image of existing applications
   3. Build script changes in graddle for containerization.
   4. Deploying application into AWS. basic introduction