**Spring Boot with Microservices**

|  |
| --- |
| **Outline**  **Day-1**   * **Spring Framework**   + Introduction to Service Layer   + Need of Spring Framework   + Introduction to Spring Framework   + Introduction to Spring IoC Container   + Configuring and using Spring IoC Container   + Java-Annotation based Configuration   + Introduction to Autowiring,   + Using @Autowired annotation and other annotations   **Day-2**   * Spring Boot   + What is Spring Boot?   + Spring starter Maven Dependencies   + Understanding @SpringBootApplication   + Example of Spring MVC-based RESTful Web Service   + Project Structure   + Externalized Configuration application.properties and YAML * Actuators   + Exposing Information about your services   + Customize Health and Info Endpoints   + Custom Metrics and Custom Actuator   + Prometheus Monitoring * Building Web Applications(MVC)   + Controllers and Model Attributes   + Template Views   + Using Embedded and External Databases   + Spring Boot Initializers and Command Line Runners   + Active Profiles   + Form Submissions   Day-3  REST Services   * + What is REST?   + Restful Controllers   + JSON and XML Data Exchange   + Content Negotiation   + Working with exception handling   + Exception handling   + Angular JS Accessing your Services * Java Clients using RestTemplate   + ResponseEntity   + Headers   + Status codes   + RequestEntity   + Posting JSON to a Service   + Integration with JMS and other backend services * Persistence with JPA Repositories   + JPA, EntityManagers, Entity Classes, Annotation mappings   + Spring JPA Data, CrudRepository, PagingAndSortingRepository   + Spring Data Rest, Exposing Databases as Endpoints   + JSON   + @Projections and Excerpts     - Restrict the data sent back to the client   + Spring Unit Testing   Day-4   * Introduction to Microservices   + What are Microservices?   + Decentralized Governance, Scalability, Fault Tolerance   + Cloud Computing   + Spring Cloud   + Microservice Tradeoffs   + Key Principles – DDD/Bounded Context   + Key Patterns – API Gateway and Security   + Third party Communication using Spring   + Service and Client Discovery   + Netflix   Day-5   * Core Microservice Patterns using Spring Cloud and Netflix OSS   + Where are my Services?     - Using Service Discovery     - Eureka Servers and Clients   + Scale Services     - Load Balancing and Service Discovery     - A LoadBalanced RestTemplate   + Circuit Breaker, when services fail     - Hystrix     - Callbacks   + Gateway/Edge Services – Providing an API     - Zuul services     - Filtering the Request and Response   + Create Feign Clients to your Services   + Messaging frameworks : Kafka     - Keeping Services synchronized with each other     - Creating Message Queues     - Project development and deployment on Spring     - Deploying to Docker container     - Microservices Security     - Microservices design patterns |
|  |