**Day 1: Introduction to Spring Boot & RESTful Services**

**Duration:** 2 hours

**Agenda / Topics**

1. Introduction to Spring Boot
   * What is Spring Boot and why it’s used
   * Spring Boot vs Spring Framework
   * Project structure and dependencies (Maven/Gradle)
2. Setting up the Environment
   * JDK, IDE (IntelliJ/Eclipse), Maven/Gradle setup
   * Spring Initializr
3. Spring Boot Annotations & Components
   * @SpringBootApplication, @RestController, @Service, @Repository
   * @Autowired and dependency injection
4. Creating REST APIs with Spring Boot
   * GET, POST, PUT, DELETE endpoints
   * Request and Response handling
5. Introduction to Exception Handling & Logging

**Lab Exercises**

* Create a Spring Boot project from Spring Initializr
* Implement a simple REST API for **Employee Management** with endpoints:
  + GET /employees
  + GET /employees/{id}
  + POST /employees
  + PUT /employees/{id}
  + DELETE /employees/{id}
* Test APIs using **Postman** or **cURL**

**Day 2: Introduction to Microservices & Spring Boot Features**

**Duration:** 2 hours

**Agenda / Topics**

1. Microservices Basics
   * Monolith vs Microservices
   * Benefits of Microservices
   * Overview of Spring Cloud
2. Building Microservices with Spring Boot
   * Creating multiple Spring Boot projects for services
   * Service layer separation
3. Configuration & Profiles
   * application.properties / application.yml
   * Profiles for dev, test, prod
4. Spring Boot Data Access
   * Using Spring Data JPA
   * Connecting to H2 / MySQL database
   * CRUD operations with JPA Repositories

**Lab Exercises**

* Create **Employee Service** and **Department Service** as separate Spring Boot projects
* Configure H2 or MySQL database for both services
* Implement CRUD operations for both services
* Use application.properties profiles for dev/test

**Day 3: Advanced Microservices Concepts & Integration**

**Duration:** 2 hours

**Agenda / Topics**

1. Inter-service Communication
   * REST API calls between services using RestTemplate / WebClient
2. Introduction to Spring Cloud / Eureka (Optional)
   * Service discovery concepts
3. Exception Handling & Validation
   * Global exception handling with @ControllerAdvice
   * Input validation with @Valid
4. Testing & Debugging
   * Unit testing with JUnit
   * API testing with Postman
5. Wrap-up & Best Practices
   * Project structure
   * Logging and monitoring

**Lab Exercises**

* Integrate **Employee Service** with **Department Service**
  + Fetch department info when retrieving employee
* Implement exception handling for invalid requests
* Write unit tests for service and controller layers
* Test inter-service calls using Postman