

# Introduction to Spring & Spring Boot

**Duration:** 1 Week

**Topics Covered:** **IOC, Beans, Dependency Injection, Maven, Annotations, Autoconfigurations, Spring\_INITIALIZER**

## What You'll Learn:

- Spring Framework Core Features like Spring IOC Container, Java-based configuration, Annotation-based configuration, Dependency Injection, and more
  - Spring Boot Fundamentals and Features – Dive into Spring Boot Internals, Autoconfiguration, Spring\_INITIALIZER, and Starter Projects
  - Internal Flow of a Spring Boot Application
  - Maven Build Tools and Lifecycle goals
- 

## Spring Boot MVC and RESTful APIs

**Duration:** 1 Week

**Topics Covered:** **Spring MVC, Web Server, REST APIs, Lombok, Model Mappers, CRUD operations, Bean Validation, Exception Handling**

## What You'll Learn:

- Spring MVC Concepts – Learn how to create APIs using MVC. Turning your API dreams into reality, one controller at a time!
  - Lombok – Simplify your Java code with Lombok
  - Setting up an In-Database Database with H2 DB
  - Bean Validation – Ensure your data is always squeaky clean before it even reaches the Service layer
  - Exception Handling in Spring Boot Applications
- 

## Spring Boot Data JPA and Database Integration

**Duration:** 1 Week

**Topics Covered:** **MySQL, DBever, JPA, Hibernate, Tools, M:N, 1:N, Database Constraints**

## What You'll Learn:

- Install MySQL database server and DBever analytics tool
- Spring Data JPA Fundamentals like data drivers, JDBC, ORM, Spring Data JPA Repositories, etc.
- Configure Hibernate ORM with MySQL Database
- Build custom queries effectively using Spring Data JPA Query Methods
- JPQL and Named Query – Master JPQL and Named Queries
- One to One, One to Many, Many to One and Many to Many Mappings with real-world examples

---

## Production Ready Spring Boot Features

**Duration:** 1 Week

**Topics Covered:** Metrics, Health Checks, Swagger Documentation, Open API, Spring Dev Tools, Logging, REST Template, Third-party APIs

### What You'll Learn:

- Spring Boot Actuator – Monitor your application's health with Spring Boot Actuator
- Spring Documentation with Swagger and Open API
- Spring Dev Tools – Supercharge your development with Spring Dev Tools
- REST Template and Third-party APIs – Integrate third-party APIs with ease. Bring the power of the internet to your app!

---

## Authentication & Authorization with Spring Security

**Duration:** 2 Weeks

**Topics Covered:** Authentication, Authorization, Google OAuth, Request Filters, JWT, Password Encoding, Session Management

### What You'll Learn:

- Create a secure Login and user registration system
- Add request filters to secure REST endpoints
- Secure against CSRF (Cross-Site Request Forgery) and XSS (Cross-Site Scripting) attacks
- Third-party registration system with Google OAuth and JWT authentication
- Implement role-based access control with Spring Boot Authorization and SQL Database
- Password Security with Encoding – Secure passwords with encoding
- Session Management – Manage user sessions effectively

---

## Spring Boot Junit Testing

**Duration:** 1 Week

**Topics Covered:** JUnit, Assert Methods, Unit Testing, Mockito

### What You'll Learn:

- Get started with testing and JUnit. Test early, test often, and test correctly
  - Dive deep into JUnit annotations and assert methods
  - Master Mockito for unit testing
-

## Spring Boot Deployment with CI/CD

**Duration:** 1 Week

**Topics Covered:** AWS, Elastic Beanstalk, Buildspec, CodeDeploy, CI/CD

### ✅ What You'll Learn:

- Use Spring Profiles to manage different environments like development, staging, and production
  - Leverage buildspec to define the build flow
  - Create Pipelines for CI/CD using AWS CodePipeline
  - Deploy with CodeDeploy on Elastic Beanstalk
- 

## Aspect-Oriented Programming (AOP)

**Duration:** 1 Week

**Topics Covered:** Cross-cutting concerns, AOP Basics, Advice types, Handling Exceptions, JoinPoints, Weaving

### ✅ What You'll Learn:

- Learn the fundamentals of Aspect-Oriented Programming (AOP) and how it improves software modularity
  - Explore different advice types (before, after, around) to manage the behavior of your application at specified join points
  - Discover the weaving process and how aspects are integrated with the main code during compile time, load time, or runtime
- 

## Caching and Database Transactions

**Duration:** 1 Week

**Topics Covered:** Redis, Spring Cache, Cache Configuration, Transaction Isolation Levels, Transactional Propagation, Database Locking Mechanisms

### ✅ What You'll Learn:

- Learn how to integrate Redis for high-performance caching in your Spring Boot applications
  - Spring Cache: Understand the Spring Cache abstraction and how to implement caching with ease using Spring Boot
  - Distributed Caching: Explore the concepts and implementation of distributed caching to enhance scalability and performance
-

## Apache Kafka in Spring Boot

**Duration:** 1 Week

**Topics Covered:** [Kafka](#), [Spring Boot Messaging](#), [Publisher](#), [Consumer](#), [Broker](#), [Kafka Schema Registry](#), [Kafka UI](#)

### ✓ What You'll Learn:

- Discover how to leverage Apache Kafka for building robust and scalable messaging systems in Spring Boot
  - Learn about the messaging capabilities in Spring Boot and how to integrate them seamlessly with Kafka
  - Explore the differences and similarities between Kafka and RabbitMQ, and learn how to choose the right messaging solution for your Spring Boot application
- 

## Microservices Architecture

**Duration:** 2 Weeks

**Topics Covered:** [Eureka Registry](#), [Microservice REST APIs](#), [Circuit Breaker](#), [Resilient4J](#), [Distributed Tracing](#), [Centralized Logs](#), [ELK Stack](#), [Zipkin](#), [Distributed DB Management](#)

### ✓ What You'll Learn:

- Learn how to use Eureka Registry for service discovery in a microservices architecture with Spring Boot
  - Explore strategies and best practices for securing microservices in a Spring Boot environment
  - Learn how to use Resilient4J to build resilient microservices with fault tolerance and latency management
  - Explore centralized logging solutions to aggregate and manage logs from all microservices
  - Master the ELK Stack (Elasticsearch, Logstash, Kibana) for powerful log analysis and visualization
  - Discover how Spring Cloud Sleuth adds tracing capabilities to microservices, integrating seamlessly with Zipkin
- 

## Docker for Spring Boot Applications

**Duration:** 1 Week

**Topics Covered:** [Docker Setup](#), [Docker Compose](#), [Dockerfile](#), [Containerization](#), [Docker Images](#), [Docker Volumes](#)

### ✓ What You'll Learn:

- Learn how to set up Docker on your development environment for efficient containerization
- Understand the usage of Docker Compose to manage multi-container applications with ease
- Gain insights into building, managing, and deploying Docker images for consistent application environments



## Kubernetes Deployment for Spring Boot Applications

**Duration:** 2 Weeks

**Topics Covered:** [Kubernetes Dashboard](#), [Deployment](#), [ReplicaSets](#), [Volumes and Volume Claims](#), [Deploying Cluster on Cloud](#), [YAML configuration](#)



### What You'll Learn:

- Learn the essentials of managing Kubernetes clusters, including setting up, scaling, and maintaining cluster resources effectively
  - Dive into the core concepts and functionalities of Kubernetes, such as Pods, Deployments, Services, and more, to orchestrate and manage containerized applications at scale
  - Explore the Kubernetes Dashboard, a web-based user interface for managing and monitoring Kubernetes clusters, to visualize and interact with cluster resources efficiently
- 



## Java Executor Framework & Scheduling

**Duration:** 1 Week

**Topics Covered:** [Java Executor Framework](#), [Schedulers](#), [Cron Jobs](#), [Task Scheduling](#)



### What You'll Learn:

- Learn Java Executor Framework to manage asynchronous operations effectively
  - Build Cron Jobs and Schedulers to manage asynchronous Operations
  - Add Task Scheduling functionalities to handle background tasks in your Spring Boot Apps
-