

## SPRING & SPRING BOOT COMPLETE ROADMAP (PHASE-WISE)

### PHASE 1 — Spring Core

#### 1. Inversion of Control (IoC)

- What is IoC container?
- BeanFactory vs ApplicationContext
- How Spring manages objects
- Object creation vs retrieval
- XML config vs Java config

#### 2. Dependency Injection (DI)

- Constructor Injection
- Setter Injection
- Field Injection (why not recommended)
- Circular dependency
- Autowiring modes (byType/byName)

#### 3. Beans & Bean Lifecycle

- @Component, @Service, @Repository
- Bean initialization
- @PostConstruct, @PreDestroy
- BeanPostProcessor
- InitializingBean, DisposableBean
- Bean scopes (Singleton, Prototype, Request, Session)

#### 4. Configuration & Bean Management

- @Configuration, @Bean
- @Lazy, @Primary, @Qualifier
- Environment abstraction
- PropertySource usage

#### 5. AOP (Aspect-Oriented Programming)

- Cross-cutting concerns
- JoinPoint, Pointcut, Advice
- Types of Advices (@Before, @After, @AfterReturning, @AfterThrowing, @Around)
- Use cases: Logging, Security, Transactions
- Proxy pattern (JDK dynamic proxy vs CGLIB)

### PHASE 2 — Spring MVC

#### 1. Architecture & Flow

- DispatcherServlet, HandlerMapping, ViewResolver

## 2. Controller Layer

- `@Controller` vs `@RestController`
- `@RequestMapping`, `@GetMapping`, `@PostMapping`
- `@PathVariable`, `@RequestParam`, `@RequestBody`, `@ResponseBody`, `ResponseEntity`

## 3. Data Binding & Validation

- DTOs
- `@Valid`, `@Validated`
- JSR-380 annotations (`@NotNull`, `@Size`)
- Custom validators

## 4. Exception Handling

- `@ExceptionHandler`, `@ControllerAdvice`
- Global exception format

## 5. Filters & Interceptors

- `HandlerInterceptor`
- `OncePerRequestFilter`
- Authentication filter

## PHASE 3 — Spring Boot

### 1. Spring Boot Basics

- Auto-configuration, Starters, Embedded Tomcat
- `CommandLineRunner`, `ApplicationRunner`

### 2. Configuration Management

- Properties, YAML
- Profiles
- `@Value`, `@ConfigurationProperties`

### 3. Logging

- Logback, `logback-spring.xml`
- Patterns, levels

### 4. Actuator

- Health, Metrics
- Custom indicators
- Prometheus + Grafana

### 5. Deployment

- JAR/WAR
- Maven/Gradle packaging
- External configuration

## PHASE 4 — JPA + Hibernate

### 1. Basics

- Entity lifecycle
- Persistence Context
- Dirty checking
- Flush modes

### 2. Mappings

- `@Entity`, `@Table`
- `@Id`, `@GeneratedValue`
- `@Column`, `@Transient`
- `@Embeddable`, `@Embedded`

### 3. Relationships

- One-to-One, One-to-Many, Many-to-One, Many-to-Many
- Cascade, Fetch types

### 4. Queries

- JPQL, Named queries
- Criteria API, Specifications
- Native SQL

### 5. Performance

- N+1 issue
- Join fetch
- DTO projection
- Caching (1st, 2nd level, EhCache/Redis)

### 6. Transactions

- `@Transactional`
- Propagation, Isolation
- Rollback rules

### 7. Migration Tools

- Flyway basics
- Versioned & Repeatable migrations

## PHASE 5 — Spring Security

### 1. Basics

- Authentication vs Authorization
- `SecurityFilterChain`
- Filter hierarchy

- UserDetailsService
- 2. Password Encoding
  - BCrypt
  - NoOp
  - Salting
- 3. Authentication Types
  - In-memory, JDBC, Custom table
  - LDAP basics
- 4. JWT
  - Structure
  - Access/Refresh tokens
  - Validation, Expiry
  - Stateless auth
- 5. Method-Level Security
  - @PreAuthorize, @PostAuthorize
- 6. CORS & CSRF
  - Global config
  - Differences

## PHASE 6 — Microservices

- 1. REST Standards
  - Idempotency
  - Pagination
  - Error formats
- 2. Communication
  - RestTemplate, WebClient, Feign Client
- 3. Service Discovery
  - Eureka Server & Client
  - Load balancing
- 4. API Gateway
  - Spring Cloud Gateway
  - Filters
  - Authentication at gateway
- 5. Config Server
  - Centralized config

- Git-based properties

- Refresh scope

## 6. Resilience4j

- Circuit breaker

- Retry, Rate limiter

- Bulkhead

- Time limiter

## 7. Distributed Logging

- Sleuth + Zipkin/Jaeger

- Traceld, SpanId

## PHASE 7 — Messaging

Kafka:

- Topics, Partitions

- Producers/Consumers

- Consumer groups

- Offsets

- Exactly-once semantics

- DLQ

- Kafka Template

- Kafka Streams basics

RabbitMQ:

- Exchanges, Queues

- Routing keys

## PHASE 8 — Cloud Deployment (AWS)

- EC2

- RDS

- S3

- IAM

- Docker

- Kubernetes basics (Pods, Deployments, Services, ConfigMaps, Secrets)

## PHASE 9 — Testing

- JUnit 5

- Mockito

- MockBean

- MockMvc

- @SpringBootTest
- TestContainers
- @DataJpaTest

#### PHASE 10 — Performance Optimization

- JVM tuning
- DB indexing, optimization
- HikariCP
- Caching
- Async processing

#### BONUS — DESIGN PATTERNS

Singleton, Builder, Factory, Proxy, Strategy, Template Method, Observer, Adapter, Repository, DTO pattern