

Department of CSE

III Year – II Sem

Enterprise Programming

(Open Elective)

Prerequisites: Core Java, Object-Oriented Programming, Basic SQL

Course Outcome: Students will learn to design, build, test, deploy, and maintain enterprise-grade Java applications using modern Java frameworks and tools.

Module 1: Java Essentials and Core Concepts 15

OOP principles, Solid Design principles, Threading, concurrency, and thread safety, Java Collections Framework, generics, and streams.

Exercises:

1. Implement a multithreaded Java program that simulates a ticket booking system ensuring thread safety.
2. Create a custom exception hierarchy for a banking application and demonstrate throwing and catching the exceptions.
3. Write a Java program to manipulate collections using streams (filter, map, reduce).

Module 2: Application Architecture and Frameworks 14

Introduction to Spring application Architecture, Inversion of Control (IoC) and Dependency Injection in Spring, Bean scopes and configuration (Java annotations), Modular architecture and clean code practices, MVC architecture using Spring MVC, Integration with templates (JSP)

Exercises:

1. Build a Spring MVC web application with JSP templates to display and add user data.
2. Demonstrate different Spring bean scopes through a sample application.

Module 3: Building APIs and Web Services 10

HTTP fundamentals, requests, and responses, RESTful API design and documentation using OpenAPI, Creating APIs using Spring Boot (GET, POST, PUT, DELETE), Consuming and producing JSON, Exception handling strategies for APIs, Versioning and security best practices for APIs

Exercises:

1. Build a Spring Boot REST API to manage a simple product catalog with CRUD operations.
2. Implement global exception handling in your API using `@ControllerAdvice`.
3. Design and document a versioned API endpoint using Swagger/OpenAPI.

Module 4: Data Persistence and Management

10

Relational databases and SQL fundamentals, JDBC programming and best practices, Object-Relational Mapping (ORM) with JPA and Hibernate, Repository patterns using Spring Data JPA, Pagination, sorting, and custom queries with JPQL

Exercises:

1. Design JPA entities for a library management system and perform CRUD operations using Spring Data JPA.
2. Write custom JPQL queries to retrieve books by author and implement pagination.

Module 5: Messaging, Security & Deployment

12

Messaging systems: JMS, ActiveMQ, Logging frameworks: SLF4J, Logback, Application caching (Spring Cache, Redis), Spring Security fundamentals: Authentication, Authorization, JWT, Containerization: Docker, CI pipeline using Jenkins and GIT operations using GitHub.

Exercises:

1. Integrate ActiveMQ messaging with a Spring Boot application to handle asynchronous processing.
2. Implement caching for frequently accessed data using Spring Cache using Redis.
3. Secure a Spring Boot REST API using JWT-based authentication.
4. Create a Dockerfile for a Spring Boot application and deploy it as a container.
5. Setup a CI/CD pipeline with GitHub Actions to build, test, and deploy your Java application.

Text books:

1. Hands-on Spring 6 and Spring Boot 3.0 by [Sagara Gunathunga](#)
Edition: 2025, bpb publications.

Reference Books:

Spring Boot 3 API Mastery by [Vincenzo Racca](#) Edition: 2025, bpb publications.

Online resources:

<https://spring.io/guides/gs/spring-boot>