Java Backend Development

Page 1: Introduction to Java Backend Development

Java Backend Development focuses on building server-side applications that handle business logic, database operations, authentication, and APIs.

Backend developers work with:
- Java Core
- Spring Framework
- Spring Boot
- REST APIs
- Databases (MySQL, PostgreSQL)
- Hibernate / JPA

Java is widely used for enterprise-level applications
because of its performance, security, and scalability.

Core Technologies

Page 2: Core Technologies

1. Core Java:
OOP concepts, collections, multithreading, exception handling.

2. JDBC:
Used to connect Java applications with databases.

3. Servlets and JSP:
Used to build dynamic web applications.

4. Maven / Gradle:
Build tools for dependency management.

5. Git:
Version control system for managing source code.

Spring & Spring Boot

Page 3: Spring Framework & Spring Boot

Spring Framework provides:
- Dependency Injection (DI)
- Inversion of Control (IoC)
- MVC Architecture

Spring Boot simplifies backend development by:
- Auto configuration
- Embedded server (Tomcat)
- Production-ready features

REST API development using:
- @RestController
- @GetMapping
- @PostMapping
- @PutMapping
- @DeleteMapping

Database & ORM

Page 4: Database & ORM

1. MySQL / PostgreSQL:
Used to store application data.

2. JPA (Java Persistence API):
ORM framework to map Java objects to database tables.

3. Hibernate:
Popular implementation of JPA.

Important Concepts:
- Entity
- Repository
- Service Layer
- Controller Layer
- CRUD Operations

Advanced Concepts & Interview Points

Page 5: Advanced Concepts

- Microservices Architecture
- JWT Authentication
- Spring Security
- RESTful Best Practices
- Exception Handling
- Logging (Log4j)
- Deployment (Docker, AWS)

Interview Important Topics:
- Difference between Spring and Spring Boot
- What is Dependency Injection?
- What is REST API?
- What is JPA vs Hibernate?
- What is Microservice Architecture?

Java Backend Development is highly in demand in IT industry.