Backend Development Mastery Roadmap

1. Languages

- Java: Enterprise-grade apps, banking, Spring Boot integration
- Python: Rapid development, Django/Flask, data-oriented
- JavaScript (Node.js): Real-time, event-driven services
- Go: High-performance microservices
- C#: Enterprise apps with .NET
- SQL: Querying relational databases

2. Frameworks

- Spring Boot (Java): REST APIs, microservices, security
- Django (Python): Full-featured, admin panel, ORM
- Flask (Python): Lightweight, more control
- Express.js (Node.js): Minimalist, fast APIs
- .NET Core (C#): Microsoft stack development

3. Databases

Relational:
- PostgreSQL: Advanced, open-source
- MySQL: Fast, popular
- SQL Server: Microsoft ecosystem
NoSQL:
- MongoDB: Document-based, flexible
- Redis: In-memory, caching
- Firebase: Real-time, mobile backend

4. ORMs

- Hibernate (Java), JPA (Java), Django ORM (Python), SQLAlchemy (Python), Mongoose (JS)

5. Core Concepts

- HTTP/REST, API Design, CRUD, MVC, Auth & Authorization, DI, DSA
- Advanced: Async, Caching, Queues, WebSockets, GraphQL, Security, Testing

6. Authentication & Security

- JWT: Stateless login
- OAuth2: Social login
- Spring Security: Secure Java apps
- bcrypt: Password hashing

7. API Tools

- Postman: Test APIs
- Swagger/OpenAPI: Docs

- Insomnia: Alternative

8. Testing Tools

- JUnit, Mockito (Java)
- PyTest (Python)
- SuperTest (Node.js)

9. Deployment & CI/CD

- Docker, Kubernetes, Jenkins/GitHub Actions
- AWS/GCP/Azure/Heroku
- Nginx, Apache

10. Version Control

- Git: Versioning
- GitHub/GitLab/Bitbucket: Collaboration

11. Libraries & Tools

- Java: Lombok, Jackson, SLF4J
- Python: Requests, Celery
- JS: Axios, dotenv, bcryptjs

12. Architecture Patterns

- Layered, Clean Architecture, Microservices, Monoliths

13. Build Tools

- Maven/Gradle (Java), npm/yarn (Node.js), pip (Python)