

JAVA PARA **MICROSSERVIÇOS** **E CLOUD: UM GUIA** **PRÁTICO**

```
void main(String[] args)  
{  
    println("kello");  
}
```



ANDERSON
AGUIAR

1ª EDIÇÃO

JAVA PARA MICROSERVIÇOS E CLOUD

Um Guia Prático

PRIMEIRA EDIÇÃO

JAVA PARA MICROSERVIÇOS E CLOUD

Um Guia Prático

Anderson Aguiar



Módulo 1: Fundamentos de Java e Programação Orientada a Objetos

1. Introdução ao curso, ferramentas e ambiente (IDE, JDK, Maven/Gradle)
 2. Fundamentos da linguagem Java
 3. Tipos de dados, operadores e controle de fluxo
 4. Programação Orientada a Objetos: classes, objetos e métodos
 5. Herança, polimorfismo, encapsulamento e interfaces
 6. Tratamento de exceções e boas práticas
-

Módulo 2: Java Moderno e APIs Essenciais

7. Java Collections Framework
 8. Java 8+ – Lambdas e Streams
 9. Trabalhando com arquivos e manipulação de dados
 10. Testes unitários com JUnit e Mockito
-

Módulo 3: Spring Framework e Spring Boot

11. Introdução ao Spring e Spring Boot
 12. Criação de APIs RESTful com Spring Web
 13. Injeção de dependência e configuração com Spring
 14. Validações com Bean Validation (Hibernate Validator)
 15. Persistência com Spring Data JPA e Hibernate
 16. Padrões DTO, Mapper e Model
 17. Testes de integração com Spring Boot Test
-

Módulo 4: Microserviços na Prática

- 18. Arquitetura de microserviços: princípios e vantagens
- 19. Projeto prático: definição de contexto e serviços
- 20. Comunicação entre serviços com REST
- 21. Comunicação assíncrona com mensageria (RabbitMQ)
- 22. Comunicação assíncrona com Kafka
- 23. Service Discovery com Spring Cloud Eureka
- 24. API Gateway com Spring Cloud Gateway
- 25. Configuração centralizada com Spring Cloud Config
- 26. Tolerância a falhas com Resilience4j
- 27. Circuit Breaker e Retry
- 28. Observabilidade: logs estruturados e tracing (Sleuth + Zipkin)

Módulo 5: Segurança e Autenticação

- 29. Introdução à segurança em microserviços
- 30. Spring Security na prática
- 31. Autenticação JWT com Spring Security
- 32. Autorização baseada em roles e escopos

Módulo 6: Cloud e DevOps

- 33. Introdução à computação em nuvem (AWS, GCP, Azure)
- 34. Docker para microserviços Java
- 35. Docker Compose para orquestração local
- 36. Deploy em cloud com AWS ECS (ou EKS se preferir)
- 37. Introdução ao Kubernetes
- 38. CI/CD com GitHub Actions ou GitLab CI

- 39. Monitoramento com Prometheus e Grafana
- 40. Logs com ELK Stack
- 41. Gerenciamento de secrets e variáveis sensíveis

Módulo 7: Projeto Final e Tópicos Avançados

- 42. Introdução ao projeto final: sistema completo com múltiplos microsserviços
- 43. Design do sistema e definição dos serviços
- 44. Implementação do serviço de autenticação
- 45. Implementação do serviço de usuários
- 46. Implementação do serviço de pedidos
- 47. Implementação do serviço de pagamento
- 48. Observabilidade e testes no sistema final
- 49. Deploy completo na nuvem (AWS ou outra)
- 50. Encerramento, revisão geral e orientações de carreira

Vivemos em uma era onde aplicações precisam ser escaláveis, resilientes e continuamente evolutivas. Nesse cenário, a arquitetura de microsserviços e a computação em nuvem deixaram de ser apenas tendências e se tornaram padrões de fato no desenvolvimento moderno. O Java, com sua robustez, maturidade e ecossistema vasto, continua sendo uma das linguagens mais relevantes para quem deseja construir soluções empresariais em larga escala.

Este livro nasce da necessidade prática de capacitar desenvolvedores — iniciantes ou experientes — a compreender e aplicar os conceitos mais modernos da programação Java voltada para microsserviços e cloud. Mais do que uma simples introdução teórica, este guia é uma jornada estruturada em 50 aulas que cobrem desde os fundamentos da linguagem até o deployment de sistemas distribuídos na nuvem, com ferramentas e tecnologias amplamente utilizadas no mercado.

Ao longo dos módulos, você será conduzido por temas essenciais como orientação a objetos, APIs modernas do Java, Spring Boot, segurança, mensageria, Docker, Kubernetes e muito mais. O objetivo é claro: transformar conhecimento em experiência prática e preparar você para os desafios reais do desenvolvimento de software distribuído na era da nuvem.

Estudar microsserviços e cloud computing com Java é investir em um futuro promissor. Seja para atuar em grandes corporações, startups inovadoras ou empreender em soluções próprias, o domínio dessas tecnologias abre portas e impulsiona carreiras.

Boa leitura e excelente jornada!

Anderson Aguiar

Rio Grande, RS

2025

Aula 1 – Introdução ao Curso, Ferramentas e Ambiente (IDE, JDK, Maven/Gradle)

1.1. Boas-vindas e objetivos do curso

Seja bem-vindo(a) ao curso **Java para Microsserviços e Cloud: Um Guia Prático**. Ao longo das próximas 50 aulas, você será conduzido em uma jornada completa, partindo dos fundamentos da linguagem Java até a construção, integração, teste e deploy de um sistema real, composto por múltiplos microsserviços na nuvem.

Este curso foi planejado para ser **prático, progressivo e alinhado às exigências do mercado**. A cada módulo, você aprenderá conceitos essenciais e aplicará esse conhecimento em projetos reais, com as principais ferramentas do ecossistema Java moderno.

Objetivos desta aula:

- Apresentar a estrutura do curso
 - Explicar o que são microsserviços e cloud computing em linhas gerais
 - Orientar a instalação e configuração do ambiente de desenvolvimento
 - Mostrar o papel das principais ferramentas: IDE, JDK, Maven e Gradle
-

1.2. Estrutura do curso

O conteúdo está dividido em 7 módulos temáticos:

1. Fundamentos de Java e POO
2. Java Moderno e APIs Essenciais

3. Spring Framework e Spring Boot
4. Microsserviços na prática
5. Segurança e autenticação
6. Cloud e DevOps
7. Projeto final com deploy na nuvem

Cada aula traz explicações teóricas e exemplos práticos para fixação. Ao final, você será capaz de:

- Desenvolver aplicações Java modernas
 - Projetar e implementar microsserviços com Spring Boot
 - Integrar serviços via REST e mensageria
 - Criar pipelines de CI/CD
 - Realizar deploy de sistemas na nuvem (AWS, GCP ou Azure)
-

1.3. O que são Microsserviços e Cloud Computing?

Antes de começarmos a codificar, é importante entender o “**porquê**” desta abordagem.

Microsserviços são um estilo de arquitetura onde o sistema é dividido em **pequenos serviços independentes**, cada um responsável por uma funcionalidade específica e comunicando-se com os demais. Isso proporciona:

- Maior escalabilidade
- Facilidade de manutenção e deploy independente
- Melhor organização de equipes e responsabilidades

Cloud computing, por sua vez, permite que esses serviços sejam executados de forma distribuída, escalável e resiliente na internet, sem necessidade de infraestrutura própria. Ferramentas como **Docker**, **Kubernetes** e plataformas como **AWS** serão nossos aliados nesta jornada.

1.4. Ferramentas que usaremos

1.4.1. IDE – IntelliJ IDEA ou VS Code

Você poderá utilizar a IDE de sua preferência, sendo as mais recomendadas:

- **IntelliJ IDEA (Community ou Ultimate):** recomendada para projetos Java, com excelente suporte ao Spring Boot.
- **Visual Studio Code:** mais leve, com extensões para Java, Spring, Docker e Kubernetes.

 **Dica:** Se você está começando agora com Java, o IntelliJ tende a ser mais amigável e completo para projetos Spring Boot.

1.4.2. JDK – Java Development Kit

Neste curso, utilizaremos o **JDK 17**, versão LTS (Long-Term Support), amplamente adotada pelo mercado.

Você pode baixá-lo de fontes confiáveis como:

- [Adoptium](#)
- [Oracle](#)

Após a instalação, verifique com:

```
bash

java -version
javac -version
```

1.4.3. Maven e Gradle – Gerenciadores de dependências

Ao trabalhar com Java, especialmente em projetos corporativos e Spring Boot, é fundamental utilizar um **gerenciador de dependências**. Este curso apresentará os dois principais:

- **Maven:** baseado em XML, com convenções bem definidas.
- **Gradle:** baseado em Groovy ou Kotlin, mais flexível e rápido.

Utilizaremos principalmente o **Maven**, mas você verá como adaptar os projetos para Gradle se desejar.

1.5. Instalando e configurando o ambiente

Passo 1 – Instalar o JDK

- Baixe e instale o JDK 17.
- Configure a variável de ambiente JAVA_HOME.

Passo 2 – Instalar a IDE

- Faça o download do IntelliJ IDEA ou VS Code.
- No caso do VS Code, instale as extensões Java Extension Pack e Spring Boot Tools.

Passo 3 – Instalar Maven e/ou Gradle

- Maven: <https://maven.apache.org/download.cgi>
- Gradle: <https://gradle.org/install/>

Dica: Ao utilizar o Spring Initializr (<https://start.spring.io>), você poderá gerar seu projeto com Maven ou Gradle automaticamente configurado.

1.6. Testando o ambiente com um projeto básico

Para garantir que tudo está funcionando, crie um projeto simples com o Spring Initializr:

1. Acesse <https://start.spring.io>
2. Escolha:
 - Project: Maven
 - Language: Java
 - Spring Boot: 3.1 ou superior
 - Group: com.seuprojeto
 - Artifact: introducao
3. Dependências: Spring Web
4. Clique em "Generate" e importe na sua IDE

Rode o projeto e verifique se aparece no console:

Started IntroducaoApplication in X seconds

1.7. Conclusão da aula

Agora que você está com o ambiente pronto, podemos mergulhar no universo Java com segurança. A partir da próxima aula, começaremos a revisar os **fundamentos da linguagem Java**, reforçando conceitos essenciais para quem quer escrever código limpo, eficiente e moderno.

Prepare-se: seu caminho rumo ao domínio de microsserviços em Java começa agora!