

# Apresentação de ferramentas e conceitos básicos

**N**esta aula, vamos explorar o ambiente de desenvolvimento e as ferramentas essenciais que utilizaremos ao longo da disciplina. O ambiente de desenvolvimento é crucial para a criação e teste de funcionalidades de software de forma segura e isolada da versão em produção. Veremos como configurar ambientes locais, de desenvolvimento, teste e homologação, garantindo a segurança e estabilidade do produto final.

Além disso, apresentaremos a linguagem de programação Java, destacando suas características robustas e versatilidade em diversas plataformas. Conheceremos o framework Spring e a ferramenta Maven, que juntos facilitam a gestão de dependências e a criação de aplicações eficientes. Para completar, exploraremos as IDEs, como o IntelliJ IDEA, e plugins que otimizarão nosso fluxo de trabalho, além de ferramentas como o Postman para testar APIs. Com esse conjunto de conceitos e ferramentas, você estará bem preparado para desenvolver projetos de software modernos e robustos.

## Ambiente de Desenvolvimento

O ambiente de desenvolvimento é fundamental para qualquer desenvolvedor. Ele consiste no espaço onde se implementam e testam funcionalidades de software sem afetar a versão em produção, garantindo a segurança do usuário final. Pode ser um smartphone, notebook ou tablet, onde o desenvolvedor cria e testa novas funcionalidades. Um ambiente de desenvolvimento permite aos desenvolvedores trabalhar em um espaço

controlado, isolado da versão em produção, evitando impactar os usuários finais.

Existem quatro tipos principais de ambientes: local, de desenvolvimento, de teste e de homologação. O ambiente local é onde o código é desenvolvido na máquina do desenvolvedor, sem ser compartilhado com outros membros da equipe. O ambiente de desenvolvimento é onde os desenvolvedores podem testar funcionalidades novas, sem controle estrito ou estabilidade garantida.

O ambiente de teste é utilizado para validar funcionalidades específicas e pode incluir um ambiente de treinamento onde usuários finais aprendem a usar o sistema sem afetar a produção. Já o ambiente de homologação é o mais próximo do ambiente de produção, usado para validações finais antes de lançar o software ao público. Finalmente, o ambiente de produção é onde o software é acessado pelo usuário final, sendo crucial mantê-lo estável e funcional para evitar prejuízos financeiros e garantir a satisfação do cliente.

## **Apresentação do Java**

Java é uma linguagem de programação amplamente utilizada em diversos tipos de produtos, incluindo aplicações web, desktop e móveis. É conhecida por ser robusta e por facilitar o desenvolvimento de sistemas complexos. Java também é uma plataforma que utiliza uma máquina virtual (JVM) para transformar o código em programas executáveis. A linguagem é fortemente tipada, o que obriga o desenvolvedor a seguir regras sintáticas e semânticas estritas, facilitando a transição para outras linguagens após seu domínio.

Existem diferentes edições do Java, como Java SE (Standard Edition) para aplicações comuns, Java EE (Enterprise Edition) para aplicações corporativas, e Java FX para aplicações audiovisuais. Java é uma linguagem multi-thread, distribuída, open source, orientada a objetos e altamente

portátil, permitindo que o mesmo código seja executado em diversas plataformas. A segurança e o dinamismo são outras características marcantes do Java, tornando-o uma escolha popular entre desenvolvedores.

Para começar a usar Java, é necessário instalar o JDK (Java Development Kit) da Oracle, configurar o ambiente de desenvolvimento e escolher a versão apropriada. A versão 17 é a mais recente e estável, utilizada em muitos projetos atuais. A instalação é simples e permite a coexistência de múltiplas versões no mesmo ambiente, facilitando a adaptação a diferentes projetos.

### **Apresentação do Spring e Maven**

O Spring é um framework para Java que facilita o desenvolvimento de aplicações, enquanto o Maven é uma ferramenta de gerenciamento de dependências. Frameworks como o Spring oferecem conjuntos de técnicas e ferramentas pré-definidas para resolver problemas específicos de desenvolvimento, tornando o processo mais ágil e eficiente.

O Spring Framework é conhecido por sua inversão de controle (IoC) e injeção de dependência, que reduzem o acoplamento do código e facilitam a manutenção. Possui vários módulos, como Spring Boot para criar aplicações web com mínima configuração, Spring Data JPA para facilitar o acesso a dados, Spring Security para implementar autenticação e autorização robustas, e Spring MVC para facilitar o desenvolvimento de aplicações web com o padrão Model-View-Controller.

O Maven, por sua vez, automatiza o gerenciamento de dependências e a construção de projetos. Utiliza o arquivo `pom.xml` para listar todas as dependências necessárias, garantindo que todas as bibliotecas e plugins estejam disponíveis para o desenvolvimento. A integração com o Spring é direta, facilitando a criação e manutenção de projetos Java.

## **IDEs, Plugins e Outros**

IDEs (Integrated Development Environments) são essenciais para o desenvolvimento eficiente. Elas combinam diversas ferramentas em uma única interface, incluindo editores de texto, compiladores e depuradores. A IDE recomendada para este curso é o IntelliJ IDEA Community Edition, uma plataforma robusta e amplamente utilizada que suporta Java e outras linguagens.

Plugins são extensões que adicionam funcionalidades à IDE, como formatação de código e verificação de erros. Eles simplificam tarefas repetitivas e melhoram a produtividade. Além da IDE, outras ferramentas importantes incluem navegadores para testar aplicações web e o Postman para testar e documentar APIs. O Postman permite criar requisições HTTP, validar respostas e automatizar testes, sendo uma ferramenta indispensável para o desenvolvimento moderno.

## **GitHub da Disciplina**

Você pode acessar o repositório da disciplina no GitHub a partir do seguinte link:

<https://github.com/FaculdadeDescomplica/Framework>. Esse espaço é o seu portal para mergulhar fundo no universo da aprendizagem interativa. Nele, você encontrará todos os códigos, além dos links para os arquivos e dados.

## **Conteúdo Bônus**

Recomendo como conteúdo bônus a documentação oficial do Spring Framework, uma fonte rica e gratuita para complementar o aprendizado. A documentação oferece guias detalhados, tutoriais e exemplos práticos

sobre a configuração do ambiente de desenvolvimento, uso do Spring Boot, Maven e muito mais.

**Título:** Spring Framework

**Plataforma:** [Spring.io](https://spring.io)

## **Referência Bibliográfica**

CARDOSO, L. da C. **Design de aplicativos**. Intersaberes: 2022

ELMASRI, R.; NAVATHE, S. B. **Sistemas de banco de dados**. 7.ed. Pearson: 2018.

JOÃO, B. do N. **Usabilidade e interface homem-máquina**. Pearson: 2017

LEAL, G. C. L. **Linguagem, programação e banco de dados**: guia prático de aprendizagem. Intersaberes: 2015.

MEDEIROS, L. F. de. **Banco de dados**: princípios e prática. Intersaberes: 2013.

PUGA, S.; FRANÇA, E.; GOYA, M. **Banco de dados**: implementação em SQL, PL/SQL e Oracle 11g. Pearson: 2013.

SETZER, V. W.; SILVA, F. S. C. **Bancos de dados**. Blucher: 2005.

VICCI, C. (Org.). **Banco de dados**. Pearson: 2014.

**Ir para exercício**