



INSTITUTO FEDERAL  
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
Bahia

---

# Linguagem de Programação II

Introdução a Frameworks em Java;



# Roteiro

- Introdução a Frameworks em Java;
  - O que é um Framework?
  - Benefícios dos Frameworks;
  - Tipos de Frameworks em Java;
  - Exemplos Populares:
    - Spring Framework;
    - Hibernate;
    - Spring MVC;
    - JUnit;
  - Vantagens de Usar um Framework;
  - Considerações e Desafios;
  - Boas Práticas ao Usar Frameworks;
  - Estudo de Caso: JUnit.



# O que é um Framework ?

---

**Uma estrutura pré-desenvolvida com bibliotecas e padrões para facilitar o desenvolvimento de software.**



# O que é um Framework ?

---

É um **conjunto de bibliotecas**, que abordam **funcionalidades**, e **estruturas**, para o **desenvolvimento de aplicações**, a fim de **fornecer soluções** para um mesmo **domínio de problema**, permitindo a **reutilização** do seu **código**.



# O que é um Framework ?

---

É formado por um conjunto classes, que fornece recursos comuns já prontos, validados e testados, os quais podem ser usados para auxiliar o desenvolvimento de software, viabilizando maior eficiência na resolução dos problemas, otimização de recursos, e detecção de erros.



# Benefícios dos Frameworks

---

- **Aceleram o desenvolvimento;**
- **Promovem boas práticas;**
- **Facilitam a manutenção.**



# Tipos de Frameworks em Java

---

- **Frameworks de Aplicação:** Gerenciam o ciclo de vida da aplicação (Inicialização, Configuração, Execução e Finalização).
- **Frameworks de Persistência:** Abstraem o acesso a bancos de dados.



# Tipos de Frameworks em Java

---

- **Frameworks Web:** Facilitam o desenvolvimento de aplicações web.
- **Frameworks de Testes:** Automatizam a execução de testes.



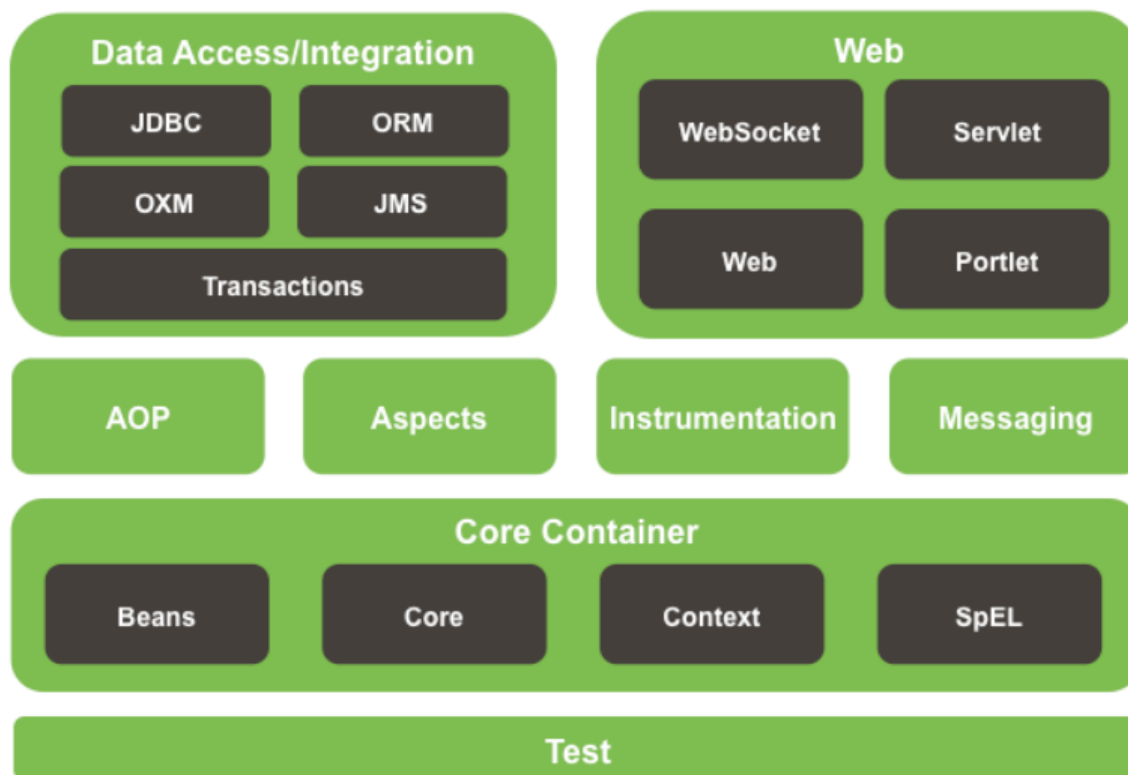


# Exemplos Populares – Spring Framework

## Arquitetura



### Spring Framework Runtime



# Exemplos Populares – Spring Framework

- **Características:**

- **Inversão de Controle (IoC):** Controle do fluxo da aplicação é transferido para o framework.
- **Injeção de Dependência:** Padrão de design em que um componente (objeto) recebe suas dependências de uma fonte externa, em vez de criá-las internamente.



# Exemplos Populares – Spring Framework

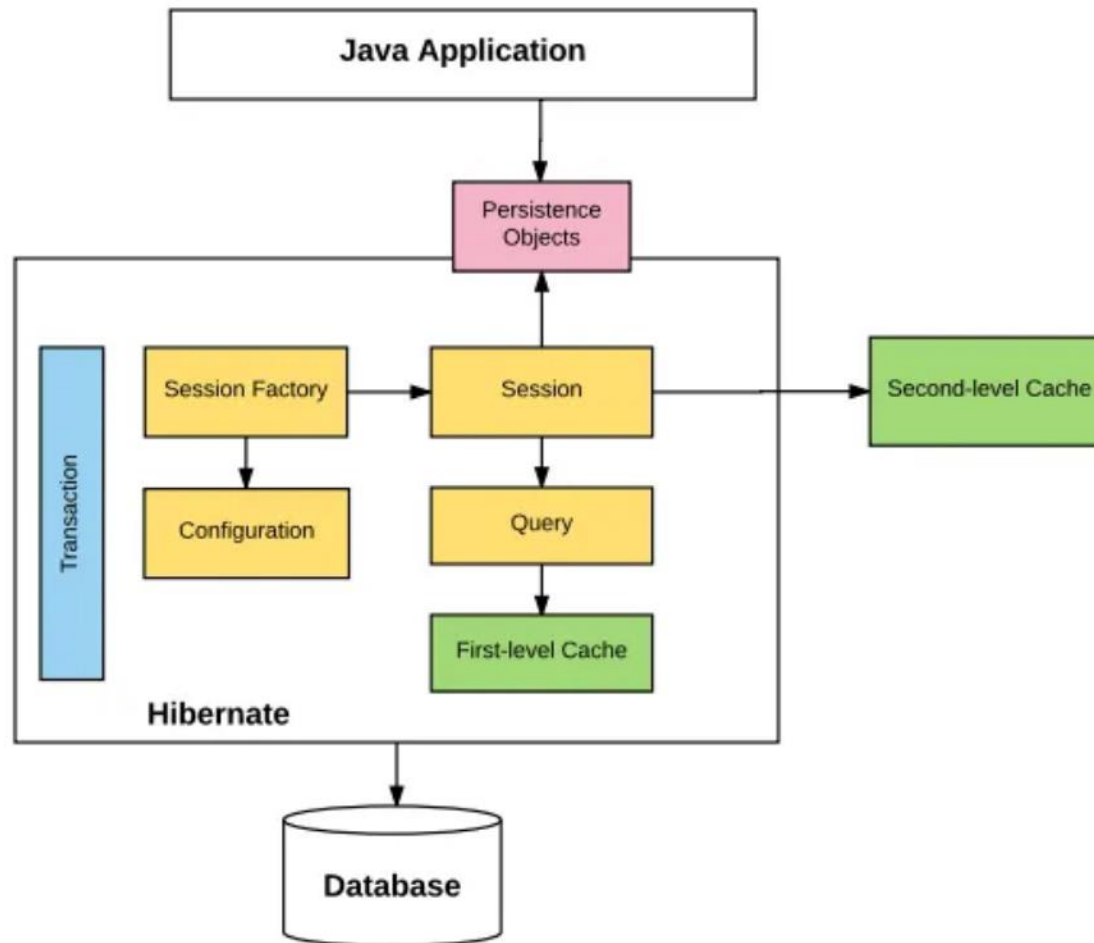
---

- **Benefício dessas Características:**
  - **Simplificam a configuração e gerenciamento de componentes (objetos).**



# Exemplos Populares – Hibernate

## Arquitetura



# Exemplos Populares – Hibernate

---

- **Característica:**
  - **Mapeamento Objeto-Relacional:** Forma de conectar objetos de uma programa orientado a objetos com tabelas de uma bando de dados relacional.



# Exemplos Populares – Hibernate

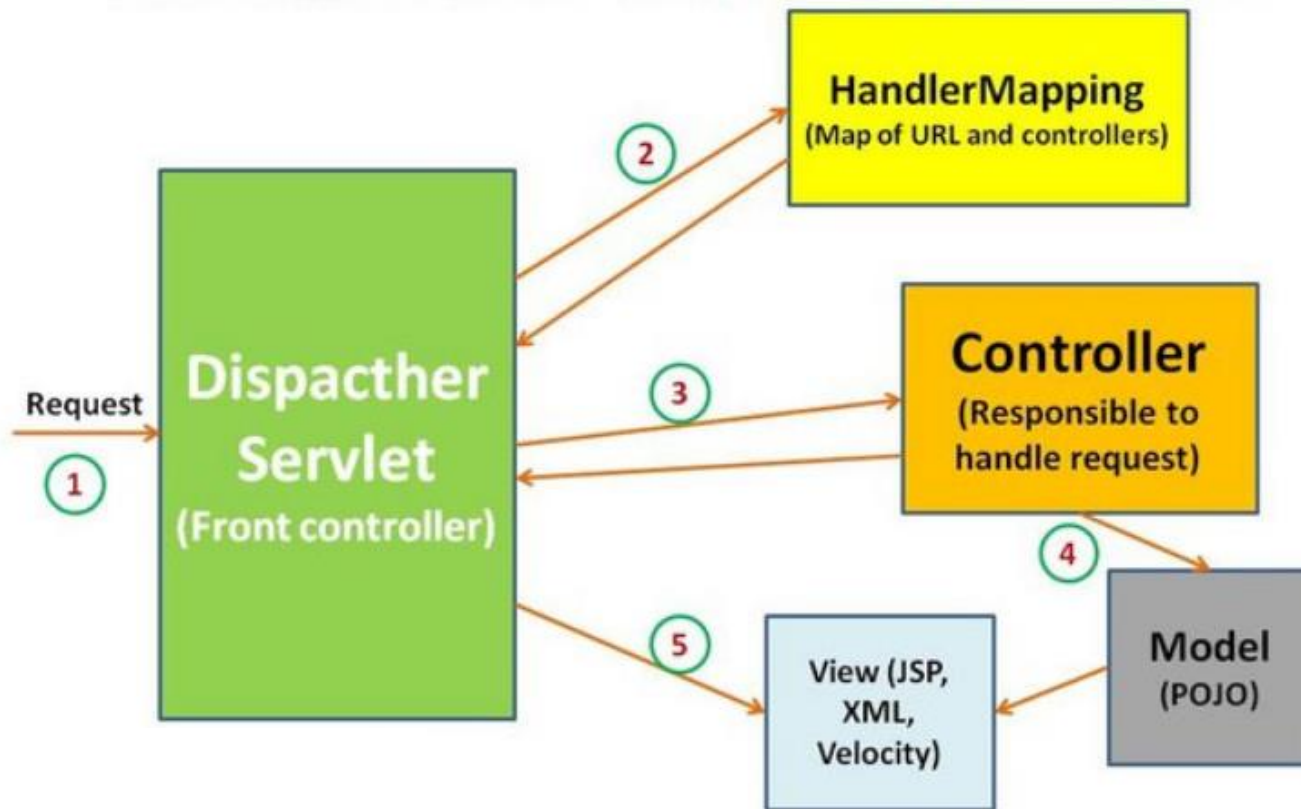
---

- **Benefício dessa Característica:**
  - **Abstrai a interação com bancos de dados.**



# Exemplos Populares – Spring MVC

## Arquitetura



# Exemplos Populares – Spring MVC

- **Característica:**
  - **Arquitetura Modelo-Visão-Controlador (MVC)** : Padrão de design que separa a aplicação em três componentes principais (**Modelo** [dados e lógica de negócio], **Visão** [Interface com o usuário] e **Controlador** [Interação entre Modelo e Visão]).





# Exemplos Populares – Spring MVC

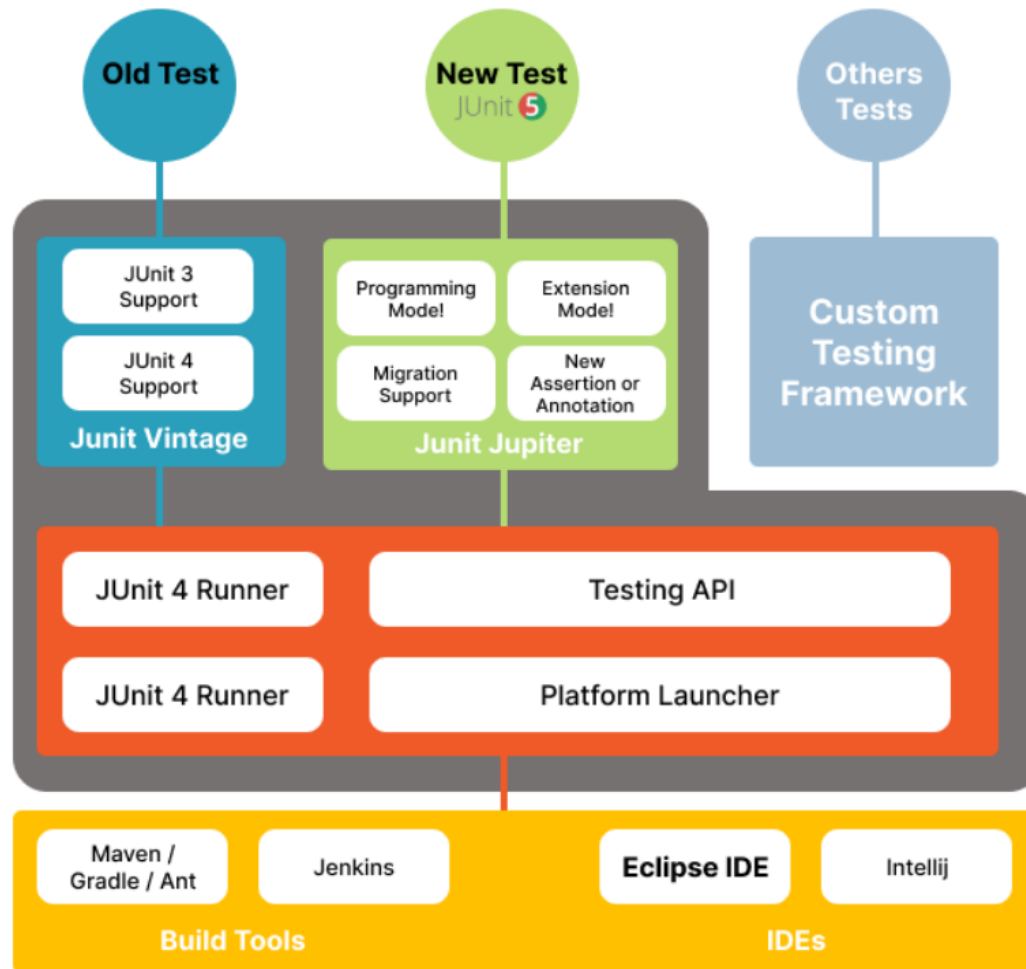
---

- **Benefício dessa Característica:**
  - **Organiza e facilita o desenvolvimento de aplicações web.**



# Exemplos Populares – JUnit

## Arquitetura



# Exemplos Populares – JUnit

---

- **Característica:**
  - **Teste Unitários** : São verificações automáticas (classes com métodos) que avaliam se partes isoladas do código (métodos) funcionam como esperado.



# Exemplos Populares – JUnit

---

- **Benefício dessa Característica:**
  - **Automatiza a verificação do comportamento correto do código.**



# **Vantagens de Usar um Framework**

---

- **Aumenta a produtividade;**
- **Padroniza o desenvolvimento;**
- **Reduz a complexidade.**



# Desafios e Considerações

---

- **Escolha adequada do framework;**
- **Curva de aprendizado;**
- **Personalização vs. Flexibilidade.**



# Boas Práticas ao Usar Frameworks

---

- **Mantenha-se atualizado com a documentação.**
- **Evite modificações profundas no código do framework.**



# Estudo de Caso - JUnit

---

- **Ver o projeto 'ProjetoTesteDeUnidade'.**





# Referências

- VALENTE, Marco Tulio. Engenharia de software moderna. Princípios e Práticas para Desenvolvimento de Software com Produtividade, v. 1, 2020.
- <https://balta.io/blog/o-que-e-um-framework>
- <https://blog.back4app.com/pt/frameworks-java/>
- <https://rollbar.com/blog/most-popular-java-web-frameworks/>



# Obrigado!

- Canais de Comunicação;
- Horário de Atendimento.

