



FIAP



Domain Driven Design using Java





AGENDA



1

Maven, SemVer e Spring Framework

2

Exercícios



Maven

O Que é o Maven?

- Definição:

- Ferramenta de gerenciamento de projetos para Java.
- Automatiza o processo de build, dependências e execução de testes.

- Principais Funcionalidades:

- Gerenciamento de dependências.
- Estruturação de projetos.
- Compilação, testes e empacotamento automatizados.

Estrutura de um Projeto Maven

```
meu-projeto/  
├── src/  
│   ├── main/  
│   │   ├── java/    # Código-fonte Java  
│   │   └── resources/ # Arquivos de configuração  
│   └── test/  # Testes unitários  
└── pom.xml  # Arquivo de configuração do Maven
```

- O **arquivo** pom.xml define::
 - Dependências.
 - Plugins.
 - Fases de build do projeto.

Fases do Maven (Lifecycle)

Fase	Descrição
validate	Valida se o projeto está correto.
compile	Compila o código-fonte Java.
test	Executa testes unitários.
package	Gera um arquivo .jar ou .war.
verify	Verifica a qualidade do código e testes.
install	Instala o artefato compilado localmente.
deploy	Publica o artefato para um repositório remoto.

Exemplo de Dependência no *pom.xml*

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
    <version>2.7.3</version>
  </dependency>
</dependencies>
```

- **groupId**: Identifica o projeto da dependência.
- **artifactId**: Nome do artefato que será baixado.
- **version**: Define a versão desejada.

Spring Framework

O Que é o Spring Framework?

- Definição:

- Framework para desenvolvimento de aplicações Java.
- Oferece integração simplificada para banco de dados, segurança e microserviços.

- Principais Módulos:

- **Spring Core:** Gerenciamento de dependências e Inversão de Controle (IoC).
- **Spring Boot:** Configuração simplificada para iniciar aplicações rapidamente.
- **Spring MVC:** Desenvolvimento de aplicações web RESTful.
- **Spring Data:** Facilita interação com bancos de dados.

Configuração Inicial de um Projeto Spring Boot

```
@SpringBootApplication
public class MeuAplicativo {
    public static void main(String[] args) {
        SpringApplication.run(MeuAplicativo.class, args);
    }
}
```

- A anotação *@SpringBootApplication* inicia a aplicação automaticamente.
- O método *SpringApplication.run* configura a aplicação sem necessidade de servidor externo.

Configuração Inicial de um Projeto Spring Boot

```
@RestController
@RequestMapping("/produtos")
public class ProdutoController {

    @GetMapping
    public List<String> listarProdutos() {
        return List.of("Notebook", "Smartphone", "Tablet");
    }
}
```

- *@RestController* transforma a classe em um controlador REST.
- *@RequestMapping("/produtos")* define a URL base do endpoint.
- *@GetMapping* retorna a lista de produtos quando acessado via **GET**.

Padrão de Versionamento SemVer

O Que é SemVer (Semantic Versioning)?

- Definição:

- Padrão de versionamento semântico usado para gerenciar versões de software.

- Formato:

```
MAJOR.MINOR.PATCH
```

- **MAJOR** (Mudança de versão principal): Alterações incompatíveis.
- **MINOR** (Melhorias e novas funcionalidades): Compatível com versões anteriores.
- **PATCH** (Correções de bugs): Pequenas mudanças sem impacto.

Exemplos de Versionamento

Versão	Descrição
1.0.0	Primeira versão estável.
1.1.0	Adição de novas funcionalidades, compatível.
1.1.1	Correção de pequenos bugs, compatível.
2.0.0	Mudança radical, pode quebrar compatibilidade.

Configuração Inicial de um Projeto Spring Boot

```
<version>1.2.3</version>
```

- 1 → Versão principal.
- 2 → Novas funcionalidades adicionadas.
- 3 → Correções de bugs.

Exercícios

Exercícios

- Crie uma aplicação SpringBoot utilizando o Spring Initializer.
- Crie uma aplicação utilizando Maven.
- Nomeie e versiona a aplicação corretamente.
- Crie uma API para listar clientes.

Referências

- **Documentação Maven:** <https://maven.apache.org>
- **Spring Framework:** <https://spring.io>
- **SemVer:** <https://semver.org>



FIAP

