Introdução ao framework Spring Boot

DIGITAL
INNOVATION
ONE

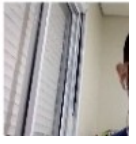
Parte 1: O que é?

Spring Boot

DIGITAL INNOVATION ONE

Problemas do Spring

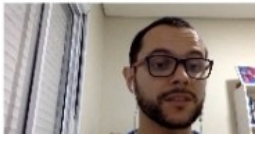
- Configurações de beans em arquivos xml.
- Dispatcher Servlet e view resolver em web.xml.
- Setup manual de Banco de Dados.
- Muito tempo gasto em configuração.
- Perda de foco em valor.



DIGITAL INNOVATION ONE

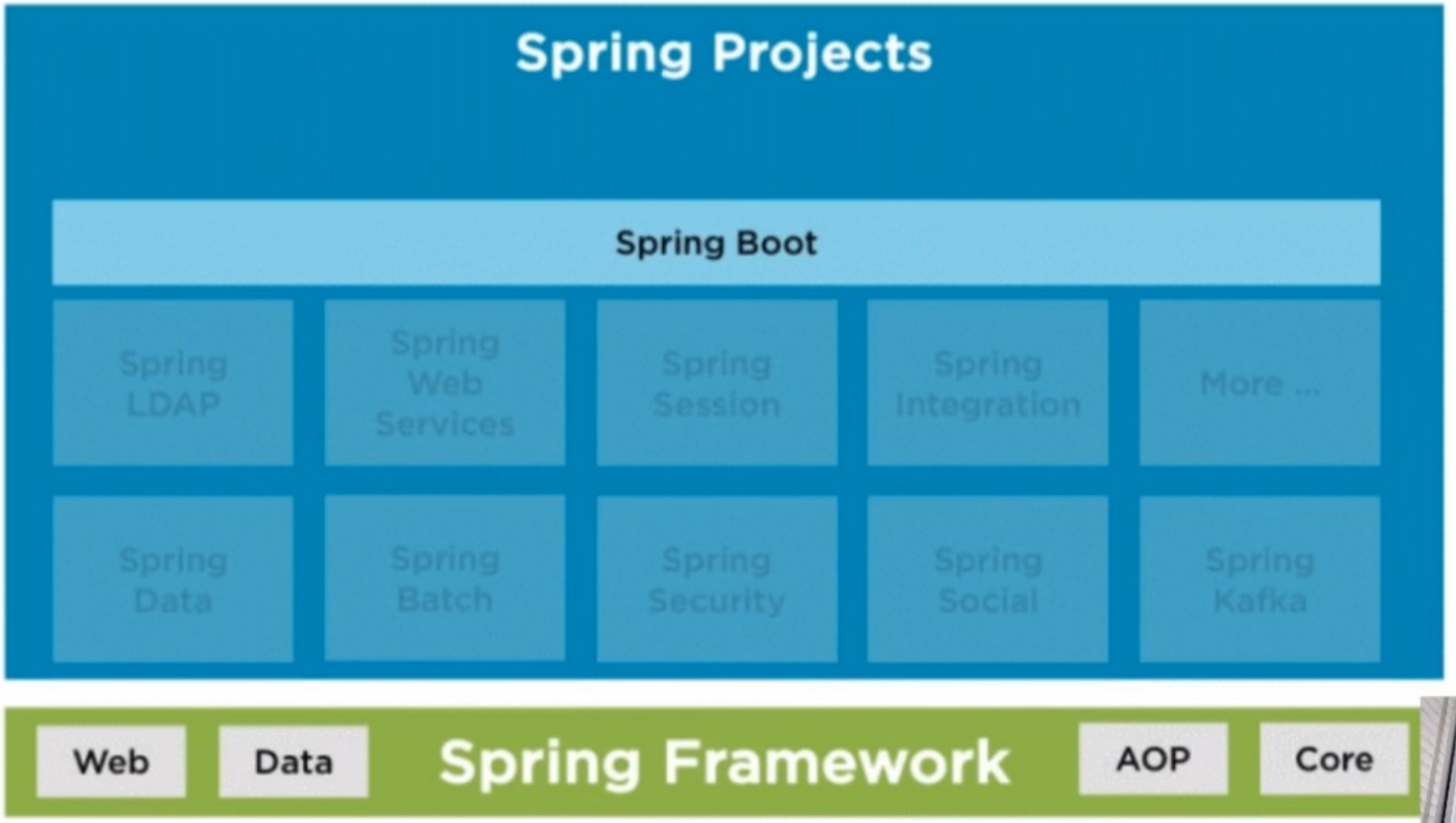
O que é o Spring Boot?

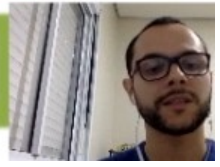
- Criado pela Spring Source em 2012.
- Facilita setup de projetos Spring.
- Sem necessidade de criar arquivos de configuração.
- Foco em produtividade.
- Maior tempo no desenvolvimento de valor.



DIGITAL INNOVATION ONE

O que é o Spring Boot?

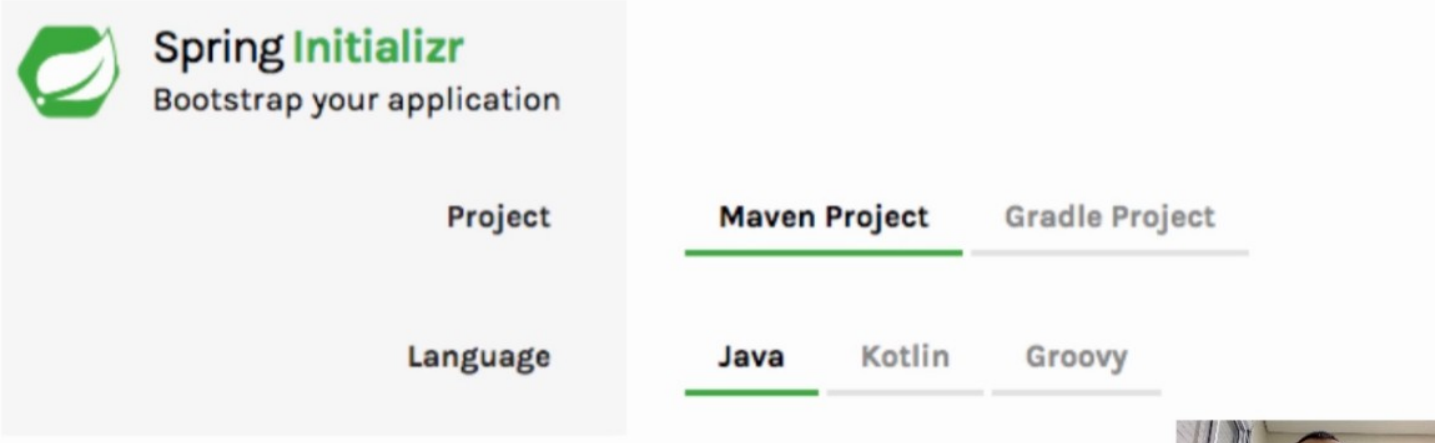


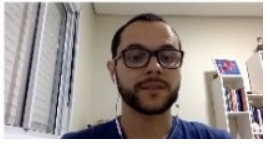


DIGITAL INNOVATION ONE

Quais problemas resolve?

- Produtividade: setup simplificado de projeto

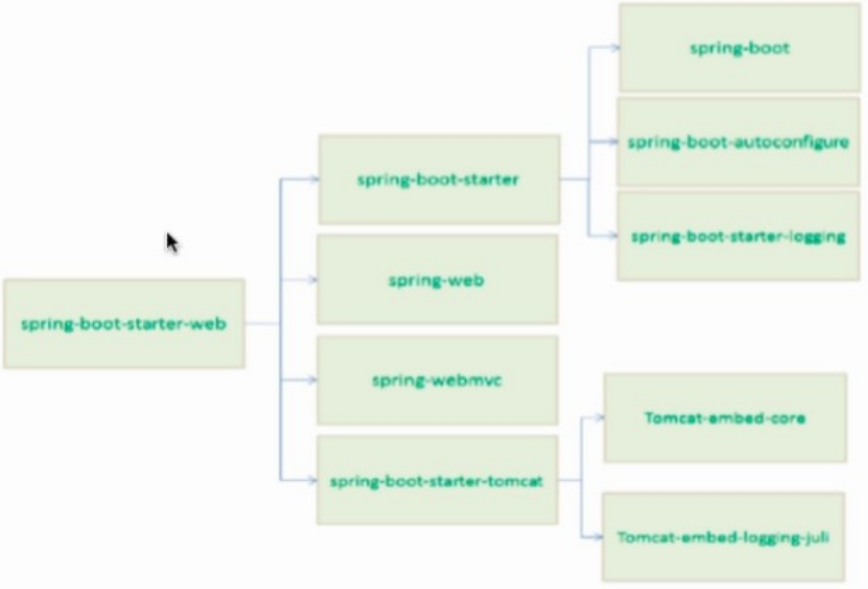


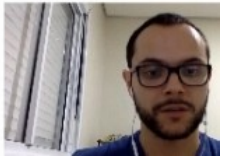


DIGITAL INNOVATION ONE

Quais problemas resolve?

Starters: dependências auto configuráveis pelo Spring Boot

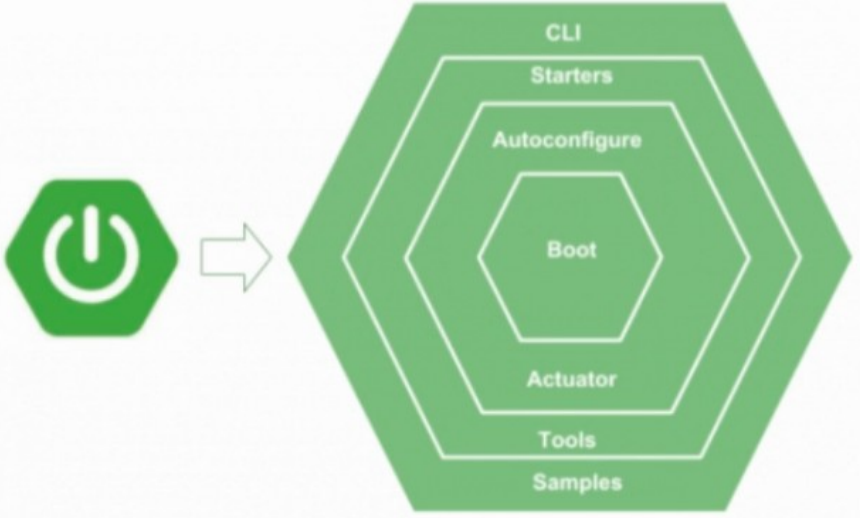


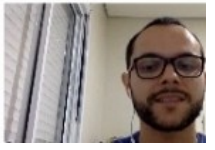


DIGITAL INNOVATION ONE

Quais problemas resolve?

- Execução simplificada: sem deploy em servidor externo.

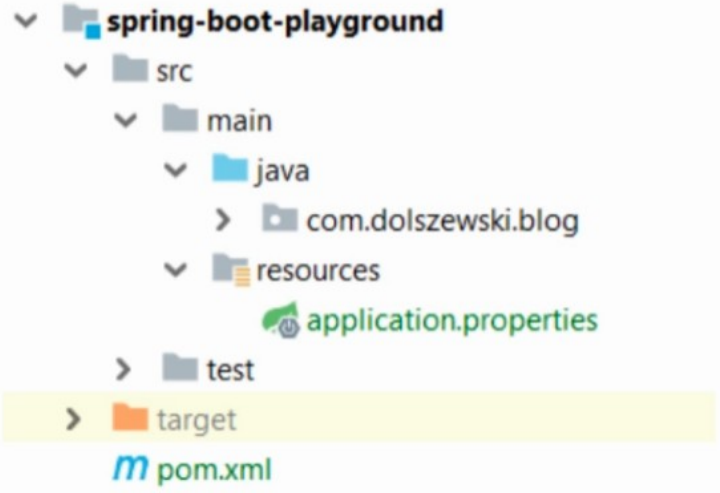


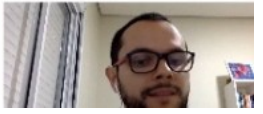


DIGITAL INNOVATION ONE

Quais problemas resolve?

- Configuração: arquivo externo para configuração.





DIGITAL INNOVATION ONE

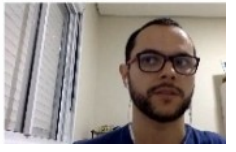
Quais problemas resolve?

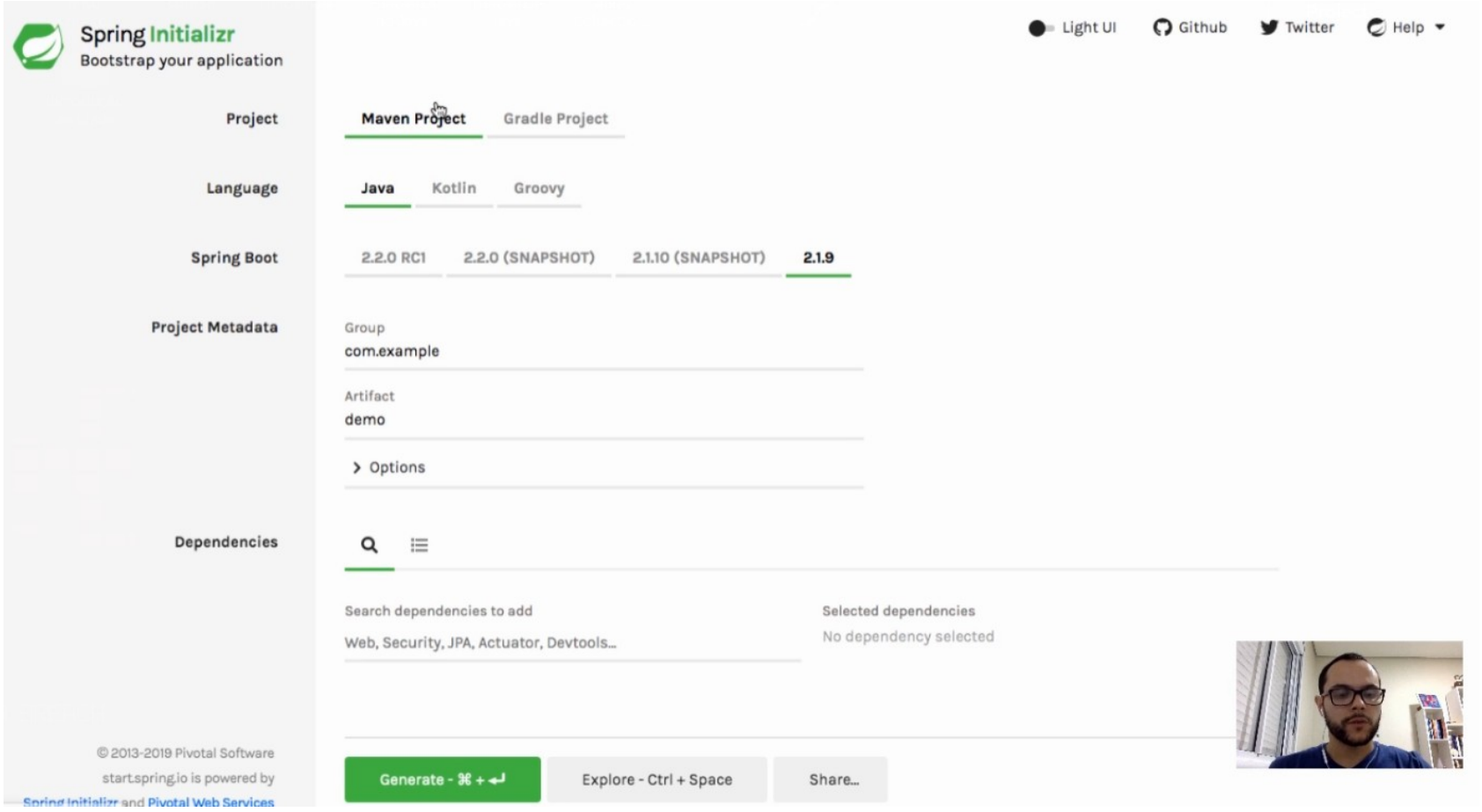
- Valor: maior tempo em desenvolvimento.

DIGITAL INNOVATION ONE

Exercício juntos

- Criação de um projeto no site http://start.spring.io.
- Importar o projeto na sua IDE favorita.
- Adicionar spring-boot-starter-mvc no pom.xml.
- Adicionar classe HelloController e o método hello().
- Executar o projeto através do terminal .




Spring Initializr
Bootstrap your application
Light UI
Github
Twitter
Help
Project
Maven Project
Gradle Project
Language
Java
Kotlin
Groovy
Spring Boot
2.2.0 RC1
2.2.0 (SNAPSHOT)
2.1.10 (SNAPSHOT)
2.1.9
Project Metadata
Group
com.example
Artifact
demo
Options
Dependencies
Search dependencies to add
Web, Security, JPA, Actuator, Devtools...
Selected dependencies
No dependency selected
© 2013-2019 Pivotal Software
start.spring.io is powered by
Spring Initializr and Pivotal Web Services
Generate - ⌘ + ↵
Explore - Ctrl + Space
Share...

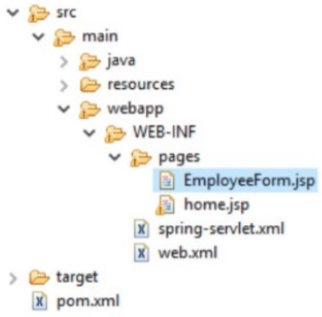
Para executar o programa usamos o comando abaixo:

```
mvn spring-boot:run
```

DIGITAL INNOVATION ONE

Configuração manual

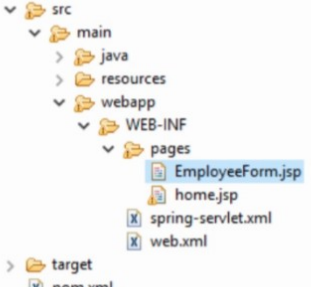
- Antes do Spring Boot...

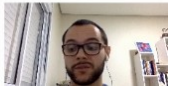


DIGITAL INNOVATION ONE

Configuração manual

- Antes do Spring Boot...

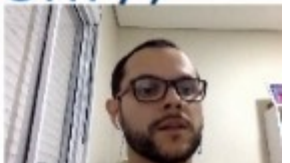




DIGITAL INNOVATION ONE

Configuração manual

- Múltiplos Arquivos XML.
- Configuração manual do Spring MVC: Dispatcher Servlet, web.xml, spring-mvc.xml.
- Configuração manual dos beans Spring.
- Aplicado também ao Spring Data, Spring Security, etc.



DIGITAL INNOVATION ONE

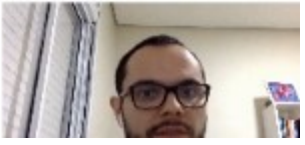
Auto configuration

- Starters: dependências simplificadas e auto configuráveis.

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

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>
        <scope>test</scope>
    </dependency>
</dependencies>
```

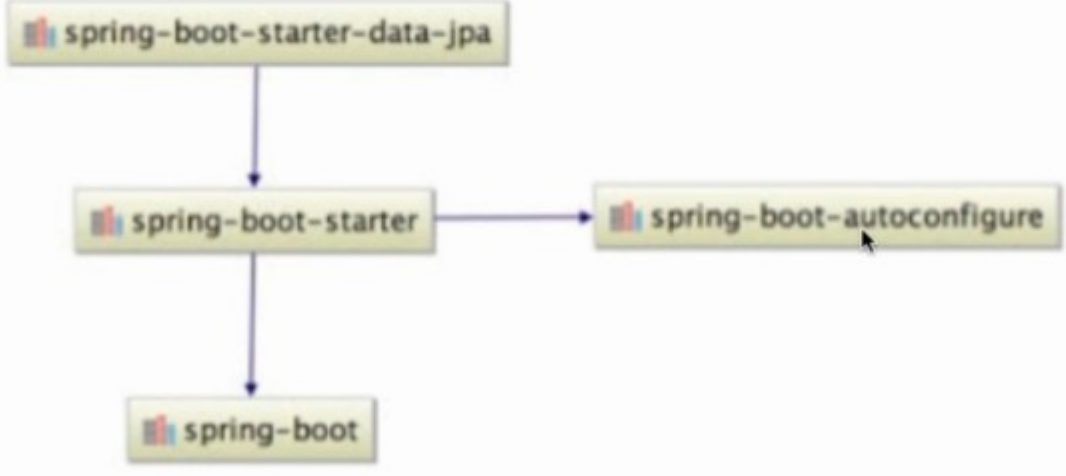
(2



DIGITAL INNOVATION ONE

Auto configuration

- Identificação e configuração automática da dependência.

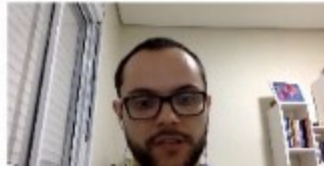


DIGITAL INNOVATION ONE

Auto configuration

- Spring Boot detecta as dependências e configura para nós!!

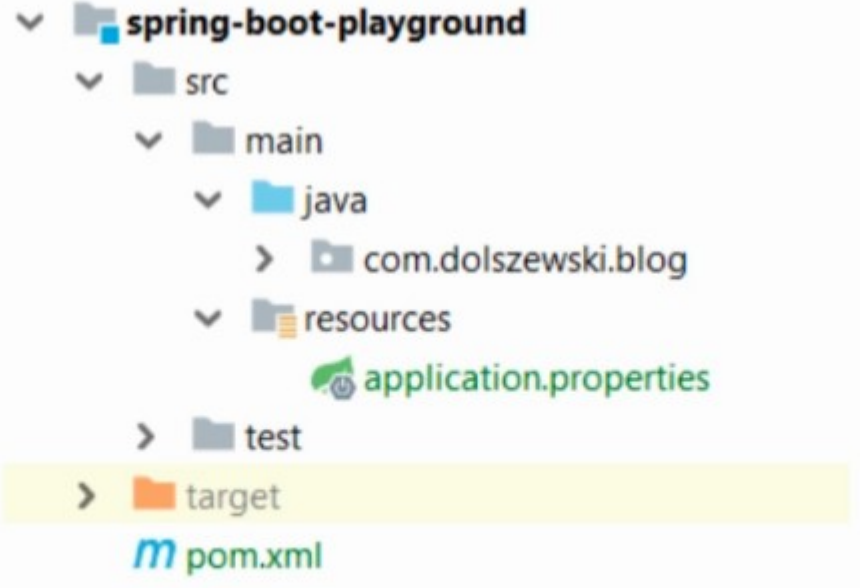
```
|    +--- META-INF
|    |    \--- spring.factories
|    +--- org.springframework.boot.autoconfigure
(...)
|    |    +--- jdbc
|    |    |    \--- DataSourceAutoConfiguration
(...)
|    |    +--- web
|    |    |    +--- DispatcherServletAutoConfiguration
|    |    |    +--- EmbeddedServletContainerAutoConfiguration
|    |    |    \--- WebMvcAutoConfiguration
(...)
|    |    +--- orm.jpa
|    |    |    +--- HibernateJpaAutoConfiguration
|    |    |    \--- JpaBaseConfiguration
|    |    \--- EnableAutoConfiguration
```

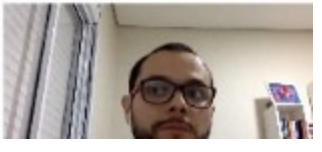


DIGITAL INNOVATION ONE

Auto configuration

- Projeto simplificado e pronto para foco no valor!!

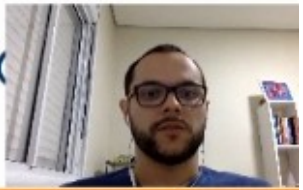




DIGITAL INNOVATION ONE

Exercício juntos

- Adicionar propriedade debug=true no application.properties.
- Executar projeto no terminal e analisar o log.
- Identificar e visualizar o auto configuration do spring mvc.
- Visualizar a dependência do auto configuratio projeto.



DIGITAL
INNOVATION
ONE

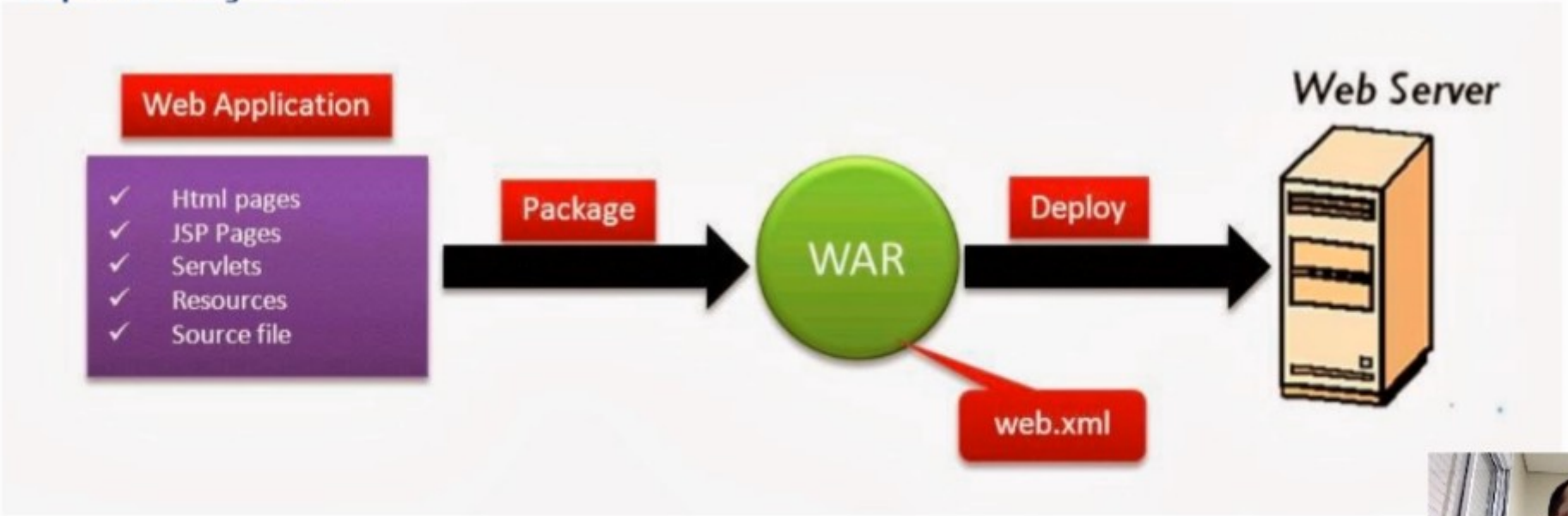
Parte 3: FatJar/ UberJar

Spring Boot

DIGITAL INNOVATION ONE

Antes do Spring Boot

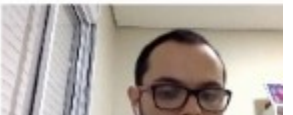
- Spring tradicional: war precisa de servidor de aplicação.



DIGITAL INNOVATION ONE

Antes do Spring Boot

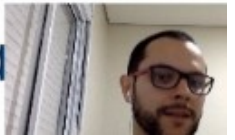
- Dependência de um container web ou servidor de aplicação.
- Complexidade para configurações.
- Atualizações frequentes, junto com versão do projeto.
- Gerenciamento manual de configurações.



DIGITAL INNOVATION ONE

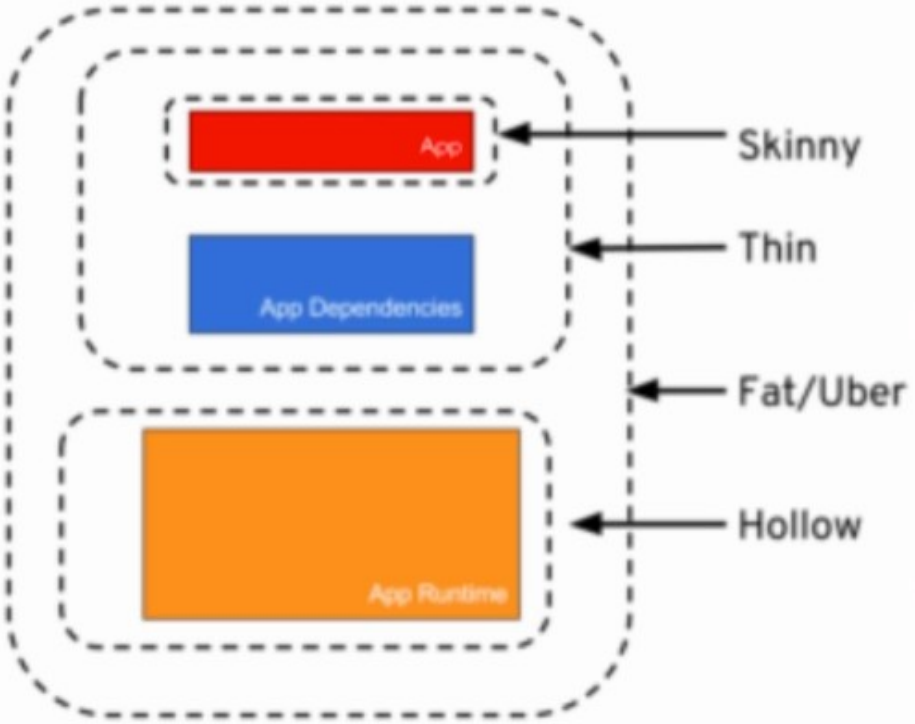
FatJar/ UberJar

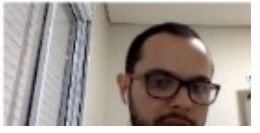
- Artefato do projeto pronto para execução.
- Container web embutido na geração e execução (Tomcat).
- Deploy embarcado com outros containers são opcionais.
- Dependências principais do projeto embarca



DIGITAL INNOVATION ONE

FatJar/ UberJar





DIGITAL INNOVATION ONE

FatJar/ UberJar

- Execução direta através de um único java -jar!!!

```
$ mvn package && java -jar target/spring-boot-example-0.1.0.jar
```

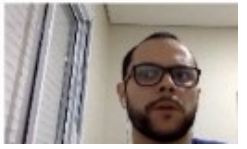
- Podemos também gerar o war tradiocional...

Create a war file

Gradle: some options in build.gradle

```
apply plugin: 'war'
war {
    baseName = 'myapp'
    version =  '0.5.0'
}
```

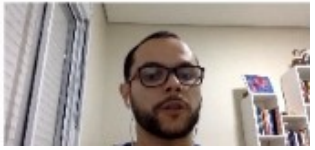
Maven: `<packaging>war</packaging>`



DIGITAL INNOVATION ONE

Exercício juntos

- Fazer o build do projeto.
- Explorar conteúdo do arquivo .jar gerado.
- Executar o projeto no terminal com java –jar.
- Trocar o formato do artefato para .war e executar no Tomcat.



Bem, ambos vão limpar. Isso significa que eles removerão a pasta de destino. A verdadeira questão é qual é a diferença entre pacote e instalação?

`package` irá compilar seu código e também empacotá-lo. Por exemplo, se o seu pom disser que o projeto é um jar, ele o criará quando você o empacotar e o colocar em algum lugar no diretório de destino (por padrão).

`install` irá compilar e empacotar, mas também colocará o pacote no seu repositório local. Isso fará com que outros projetos possam se referir a ele e capturá-lo no seu repositório local.

`mvn clean package`	para empacotar o projeto
java -jar springboot-0.0.1-SNAPSHOT.jar	para rodar o projeto empacotado.
jar tf springboot-0.0.1-SNAPSHOT.jar | less	Serve para listar e exibir todo o conteúdo do arquivo .jar

DIGITAL INNOVATION ONE

Exemplos da aula

- http://github.com/rpeleias/springboot_digital_innovation_one.

DIGITAL INNOVATION ONE

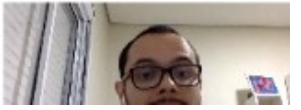
Referências

https://dzone.com/articles/spring-boot-framework-tutorials

https://www.javaavancado.com/o-que-e-spring-boot/

https://www.tutorialspoint.com/spring_boot/spring_boot_introduction.htm

https://docs.spring.io/spring-boot/docs/2.2.0.M5/reference/html/spring-boot-features.html



O que é o Spring Boot?

O auto configuration é um dos principais recursos do Spring Boot: detecta e configura nossas dependências de forma transparente para nós. Como habilitamos esse poderoso recurso em nosso projeto?

- Adicionamos a dependência spring-boot-auto-configure no arquivo pom.xml.
- Nenhuma das alternativas está corretas.
- Não precisamos fazer nada. A anotação @SpringBootApplication automaticamente adicionada para nós o recurso do auto configuration. ✔
- Todas as alternativas estão corretas.
- Precisamos adicionar a anotação @EnableAutoConfiguration no arquivo main gerado automaticamente no setup do nosso projeto.

PRÓXIMA PERGUNTA

O que é o Spring Boot?

Uma das grandes vantagens do Spring Boot é, ao fazer o build do projeto, ter à disposição um projeto pronto para deploy em ambientes de teste, desenvolvimento e/ou produção. Por default, é gerado um arquivo no formato .jar. Como fazemos para executar, de forma simples e direta, este arquivo gerado?

- Através de um simples comando java -jar <nome_do_arquivo_gerado>.jar ✔
- Renomear o arquivo .jar em um arquivo .war e fazer o deploy e um container como o Tomcat.
- Nenhuma das alternativas.
- Transferir o arquivo gerado pelo build para um servidor de aplicação, e fazer o deploy.
- Criar um arquivo executável .exe no Windows e .sh no Linux e executá-lo de acordo com o sistema operacional.

PRÓXIMA PERGUNTA

O que é o Spring Boot?

Como é conhecido o arquivo gerado após o build do nosso projeto Spring Boot, que engloba todas dependências de projeto e está pronto para a execução ?

- War; ✖
- Um simples arquivo .zip é gerado.
- fat/uber jar; ✔
- Thin jar;
- Ear;

PRÓXIMA PERGUNTA

O que é o Spring Boot?

Ao adicionar a dependência spring-boot-starter-web no nosso projeto, um container para deploy da aplicação web é adicionado automaticamente no projeto. Qual é este container?

- Tomcat. ✔
- Undertow.
- Wildfly.
- Jetty.
- Glassfish.

PRÓXIMA PERGUNTA

O que é o Spring Boot?

Ao adicionar a dependência spring-boot-starter-web, por padrão o projeto executado pelo Spring Boot sobre na porta 8080. Como é possível alterar esta porta?

- Direto na implementação do Controller.
- Através da propriedade server.port no arquivo application.properties. ✔
- Através da propriedade spring.boot.server.port no arquivo application.properties.
- Direto no arquivo pom.xml. ✖
- Nenhuma das alternativas está correta.

PRÓXIMA PERGUNTA

O que é o Spring Boot?

Qual o padrão de nomenclatura que o Spring Boot introduz para simplesmente, ao adicionar dependências no nosso arquivo pom.xml, ter a nossa disposição a dependência pronta para uso?

- spring-boot-init;
- spring-boot-starter; ✔
- spring-boot-setup;
- spring-boot-initializer;
- spring-file-setup;

PRÓXIMA PERGUNTA

O que é o Spring Boot?

Qual dos tópicos abaixo não é verdadeiro sobre o Spring Boot?

- Disponibiliza como padrão o tomcat como container web ao adicionar a dependência spring-boot-starter-web.
- Implementa o recurso do auto configuration.
- Gera automaticamente um arquivo war como executável. ✔
- Não são necessárias configurações do tipo XML, como em projetos Spring tradicionais.
- Disponibiliza starters como dependência.

PRÓXIMA PERGUNTA

O que é o Spring Boot?

Ao subir nosso projeto Spring Boot é possível ver no log como o Auto Configuration detecta um recurso auto configurável. Como fazer para habilitar este recurso de visualização do log?

- Nenhuma das alternativas está correta.
- Nada. O auto configure é automaticamente exibido no terminal.
- Passamos o argumento debug=true junto com a execução do nosso projeto através do terminal.
- Configurar o logging da aplicação a nível de DEBUG.
- Adicionar a propriedade debug e atribuímos o valor true no arquivo application.properties ✔

PRÓXIMA PERGUNTA

O que é o Spring Boot?

Quais conjuntos de anotações definem uma classe que faz parte do auto configuration?

Classes anotadas com @EnableAutoConfiguration.

Classes com a anotação @Conditional e nível de classe, seguida das anotações do conjuntos de @Conditional (@ConditionalOnClass, @ConditionalonMissingBean, @ConditionalOnProperty). ✔

Nenhuma das alternativas.

Nada. O auto configuration reconhece todas as classes dos frameworks adicionados como dependências como auto configuráveis. ✖

Nenhuma. Uma classe auto configurável deve ser informada através do arquivo de configuração application.properties.

PRÓXIMA PERGUNTA

O que é o Spring Boot?

Qual dos problemas abaixo o Spring Boot resolve?

Disponibiliza, de forma automática, arquivos de configuração xml para configurar o DispatcherServlet do nosso projeto web.

Criação de entidades: disponibiliza automaticamente baseado em convenções, templates para criação de entidades e seus respectivos DAOs. Basta apenas executar um comando no terminal e passar os templates como referência.

Todas as alternativas. ✖

Testabilidade: o Spring Boot detecta todos os trechos de código do nosso sistema que não há cobertura de testes e, automaticamente, gera código que faça esta cobertura.

Produtividade: disponibiliza, desde o começo, um projeto pronto para desenvolvimento e execução. Nenhuma configuração extra de arquivos web, banco de dados e suporte a transações é requerido. ✔

FINALIZAR