

# Spring Boot

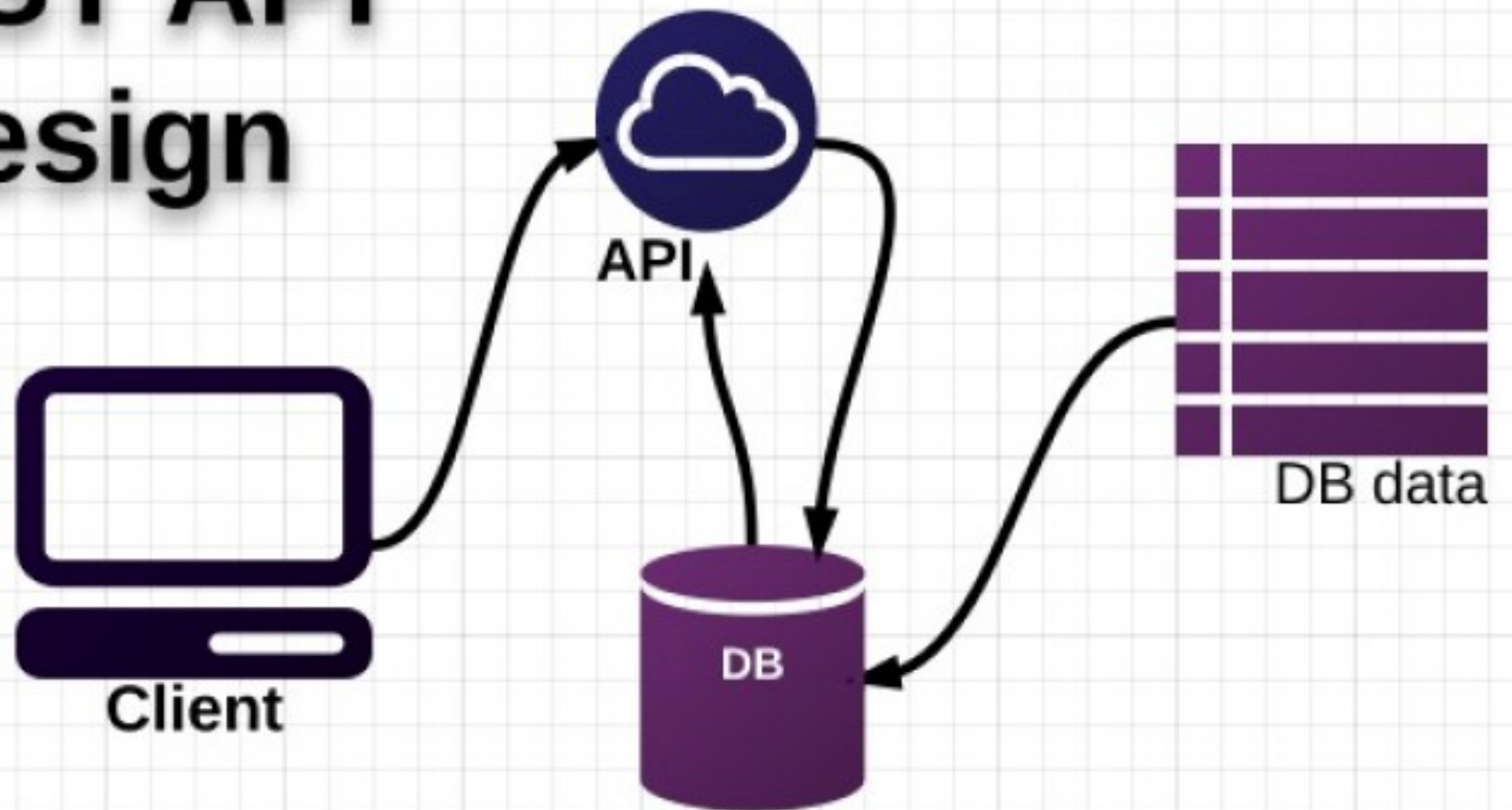
Criando uma API com Spring Boot

# Agenda

- O que são APIs?
- Spring Boot
- O Projeto
- Maven
- Lombok
- Entidades e Repositórios
- Bora codar...
- Próximos passos

# REST API Design

GET	/tasks - display all tasks
POST	/tasks - create a new task
GET	/tasks/{id} - display a task by ID
PUT	/tasks/{id} - update a task by ID
DELETE	/tasks/{id} - delete a task by ID



O que são APIs?

# Spring Boot

- Embed Server (Tomcat)
- Construído para sua aplicação estar no ar o mais rápido possível
- Convention over configuration (bye by XML hell)
- Make jar not war
- Usado por empresas como Netflix em sua arquitetura de microservices.
- Tecnologia a ser adotada: <https://www.thoughtworks.com/pt/radar/languages-and-frameworks>



# O Projeto

- O objetivo do projeto vai ser construir uma API para uma agenda de contatos.
  - Criar um contato
  - Deletar um contato
  - Atualizar um contato
  - Obter todos os contatos
  - Obter 1 contato
  - Obter contatos por um filtro

# Verificando ambiente

- Acesse o terminal ou cmd e digite os comandos abaixo:
  - `mvn -version` // verifica se o maven está instalado
  - `java -version` // verifica se o java está instalado
  - `javac -version` // verifica se o compilador do java está instalado
- Verifique se uma das IDEs indicadas estão instaladas.
  - Eclipse
  - IntelliJ
  - Netbeans

Generate a Maven Project with Spring Boot 1.3.3

## Project Metadata

Artifact coordinates

Group

br.edu.puccampinas.cursospring

Artifact

curso-spring

## Dependencies

Add Spring Boot Starters and dependencies to your application

Search for dependencies

Web, Security, JPA, Actuator, Devtools...

Selected Dependencies

Web ×

JPA ×

Generate Project ✖ + ↺

Don't know what to look for? Want more options? [Switch to the full version.](#)

# Criando o Projeto

<https://start.spring.io/>

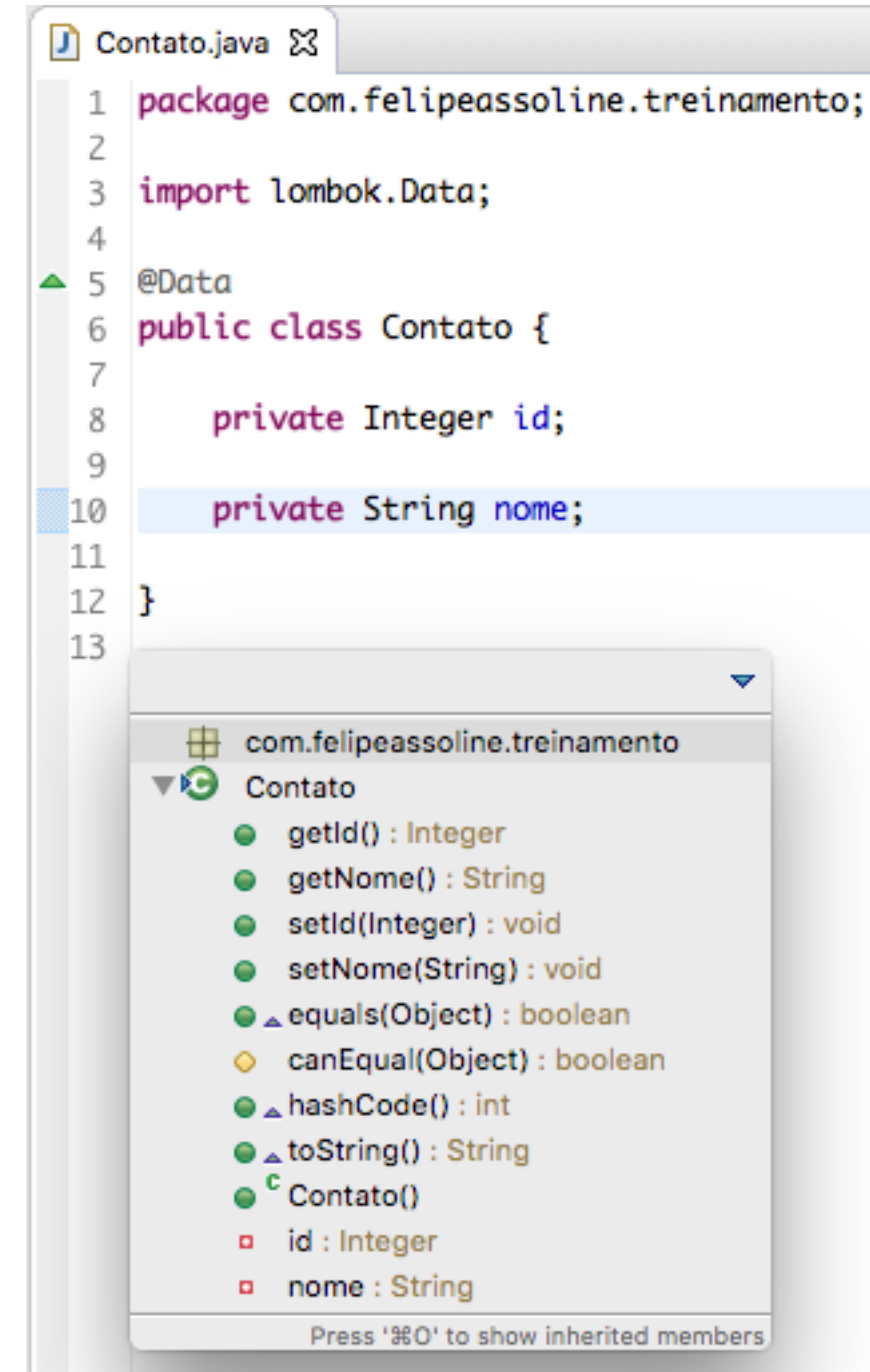


- O Maven é uma ferramenta desenvolvida pela Apache, ela serve para gerenciar as dependências e automatizar seus builds.
- pom.xml
- Dependências e “Sub-Dependências”.
- Build
- Test
- Deploy



# Lombok

- <https://projectlombok.org/>
- Basicamente cria para você os Getters e Setters da sua classe.
- Também pode ser usada para outros métodos “chatos” de criar, como toString, equals, hashCode e etc.
- @Data faz todos os métodos
- Deixa a sua classe mais “clean”



```
1 package com.felipeassoline.treinamento;
2
3 import lombok.Data;
4
5 @Data
6 public class Contato {
7
8     private Integer id;
9
10    private String nome;
11
12 }
13
```

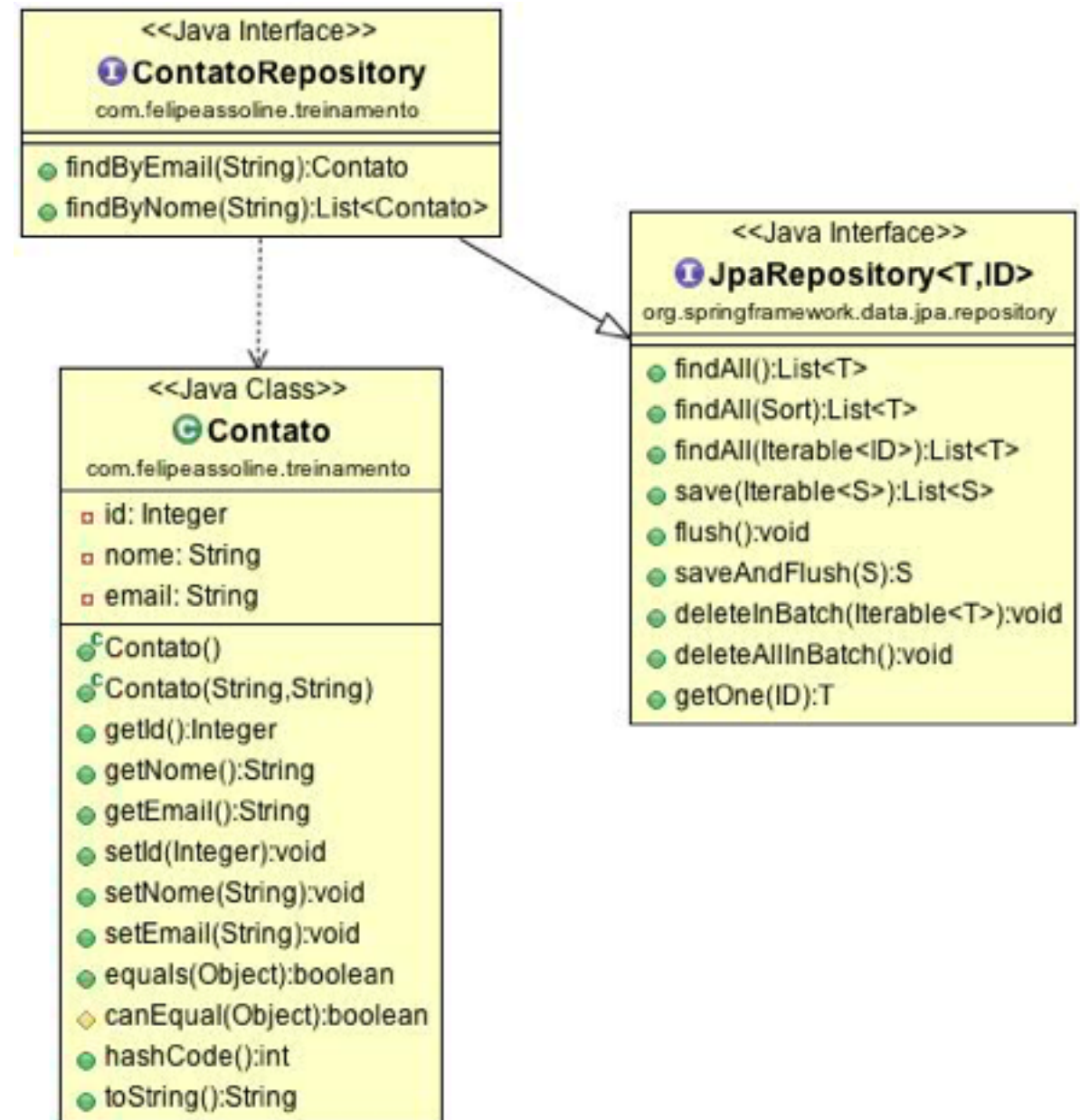
The screenshot shows a Java IDE window titled 'Contato.java'. The code defines a package, imports 'lombok.Data', and defines a 'Contato' class with two private fields: 'id' (Integer) and 'nome' (String). The '@Data' annotation is applied to the class. Below the code, a pop-up window displays the methods generated by Lombok for the 'Contato' class, including getters, setters, equals, hashCode, toString, and a constructor. The fields are also listed at the bottom of the pop-up.

Method	Return Type
getId()	Integer
getNome()	String
setId(Integer)	void
setNome(String)	void
equals(Object)	boolean
canEqual(Object)	boolean
hashCode()	int
toString()	String
Contato()	
id	Integer
nome	String

Press '%%O' to show inherited members

# Entidades e Repositórios

- Spring Data / JPA
- Convenção de nomes para métodos de repositório
- Lombok



# Bora codar...

- Download do projeto base:
- <http://bit.ly/1XtcXR9>
- Apenas os testes, entidades e repositórios estão implementados.
- Nossa tarefa será criar os endpoints do serviço.



# Próximos passos

- Adicionar camada de segurança
- Fazer deploy da aplicação

# Dúvidas?

[fassoline@gmail.com](mailto:fassoline@gmail.com)