



Desenvolvimento de APIs REST com Spring Boot

Curso

Professor: Adriano Lima



Agenda

- Noções teóricas básicas
 - REST
 - Rotas e Verbos
 - Tipos de Parâmetros
- Spring
- Padrões
- Construção do CRUD
- Configuração do Banco
- Postman
- Tratamento de Exceptions

Review

- **Representational State Transfer (REST)** é um estilo arquitetônico aplicado para fornecer padrões entre sistemas web, facilitando a comunicação entre eles.
- Maior interoperabilidade.
- Pode diminuir custos para manter/evoluir.

Rotas e Verbos HTTP

- As REST APIs fornecem uma url base (rota), onde os verbos HTTP irão indicar qual ação está sendo requisitada pelo consumidor do serviço.

Rotas	Métodos	Ações
/product	GET	Retorna a lista de produtos
/product	POST	Cria um novo produto
/product/{id}	GET	Retorna o produto com id = {id}
/product/{id}	PUT	Substitui os dados do produto com id = {id}
/product/{id}	DELETE	Remove o produto com id = {id}

Tipos de Parâmetros

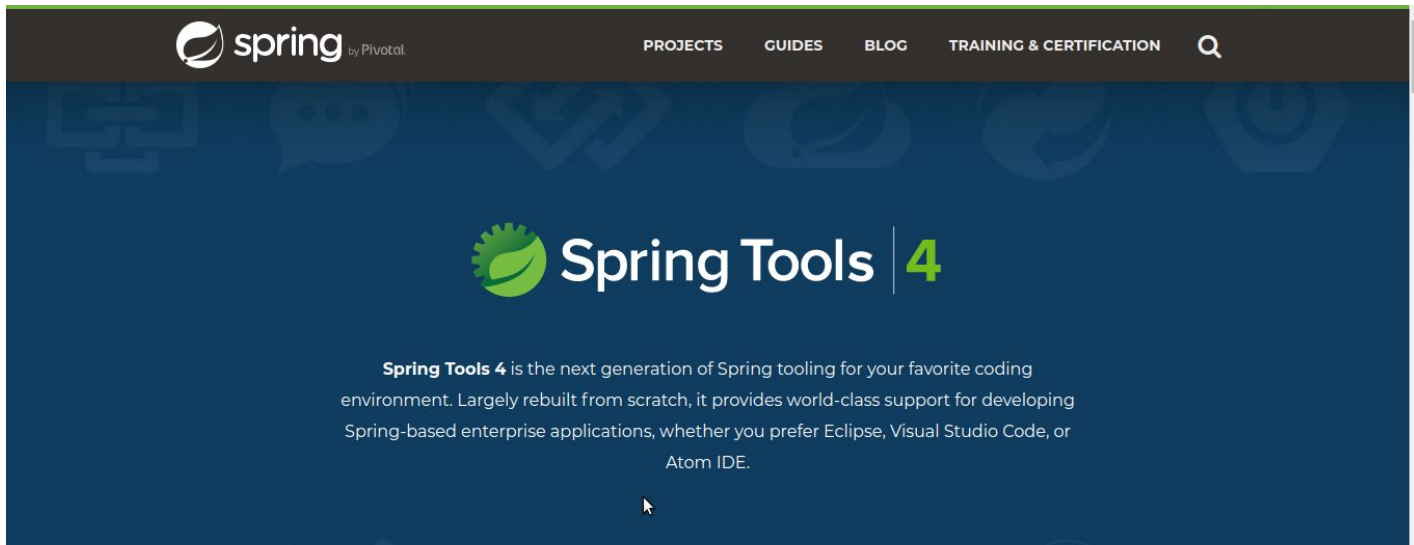
- Comumente são vistos quatro tipos de parâmetros nas APIs REST, sendo eles:
 - Header Parameters - Ex.: `sessionId: 258dsf5ad8d`
 - Query Parameters - Ex.: `/users?role=admin`
 - Path Parameters - Ex.: `/users/{id}`
 - Body Parameters - Ex.: `{"name": "Josias", "email": "josias@mail.com"}`

Tecnologias ~ Spring

- Diversas tecnologias e Frameworks.
- O Spring objetiva velocidade, simplicidade e produtividade na construção de APIs.

Instalação

- IDE ~ <https://spring.io/tools>
- Como opcional, utilize a ferramenta do spring para a criação dos projetos
 - <https://start.spring.io/>





Project

☒ Maven Project

☐ Gradle Project

Language

☒ Java ☐ Kotlin

☐ Groovy

Spring Boot

☐ 3.0.0 (SNAPSHOT) ☐ 3.0.0 (M3)

☐ 2.7.1 (SNAPSHOT) ☒ 2.7.0 ☐ 2.6.9 (SNAPSHOT)

☐ 2.6.8

Project Metadata

Group

Artifact

Name

Dependencies

ADD DEPENDENCIES... ⌘ + B

Spring Web

WEB

Build web, including RESTful, applications using Spring MVC.
Uses Apache Tomcat as the default embedded container.

Spring Boot DevTools

DEVELOPER TOOLS

Provides fast application restarts, LiveReload, and configurations
for enhanced development experience.

Spring Data JPA

SQL

Persist data in SQL stores with Java Persistence API using
Spring Data and Hibernate.

PostgreSQL Driver

SQL

A JDBC and R2DBC driver that allows Java programs to
connect to a PostgreSQL database using standard, database

GENERATE ⌘ + ↵

EXPLORE CTRL + SPACE

SHARE...



Project Metadata

Group br.edu.unichristus

Artifact aularest

Name aularest

Description Construção de APIs REST com SpringBoot

Package name br.edu.unichristus.aularest

Packaging ☒ Jar ☐ War

Java ☐ 18 ☒ 17 ☐ 11 ☐ 8

Spring Data JPA SQL

Persist data in SQL stores with Java Persistence API using Spring Data and Hibernate.

PostgreSQL Driver SQL

A JDBC and R2DBC driver that allows Java programs to connect to a PostgreSQL database using standard, database independent Java code.

GENERATE ⌘ + ↵

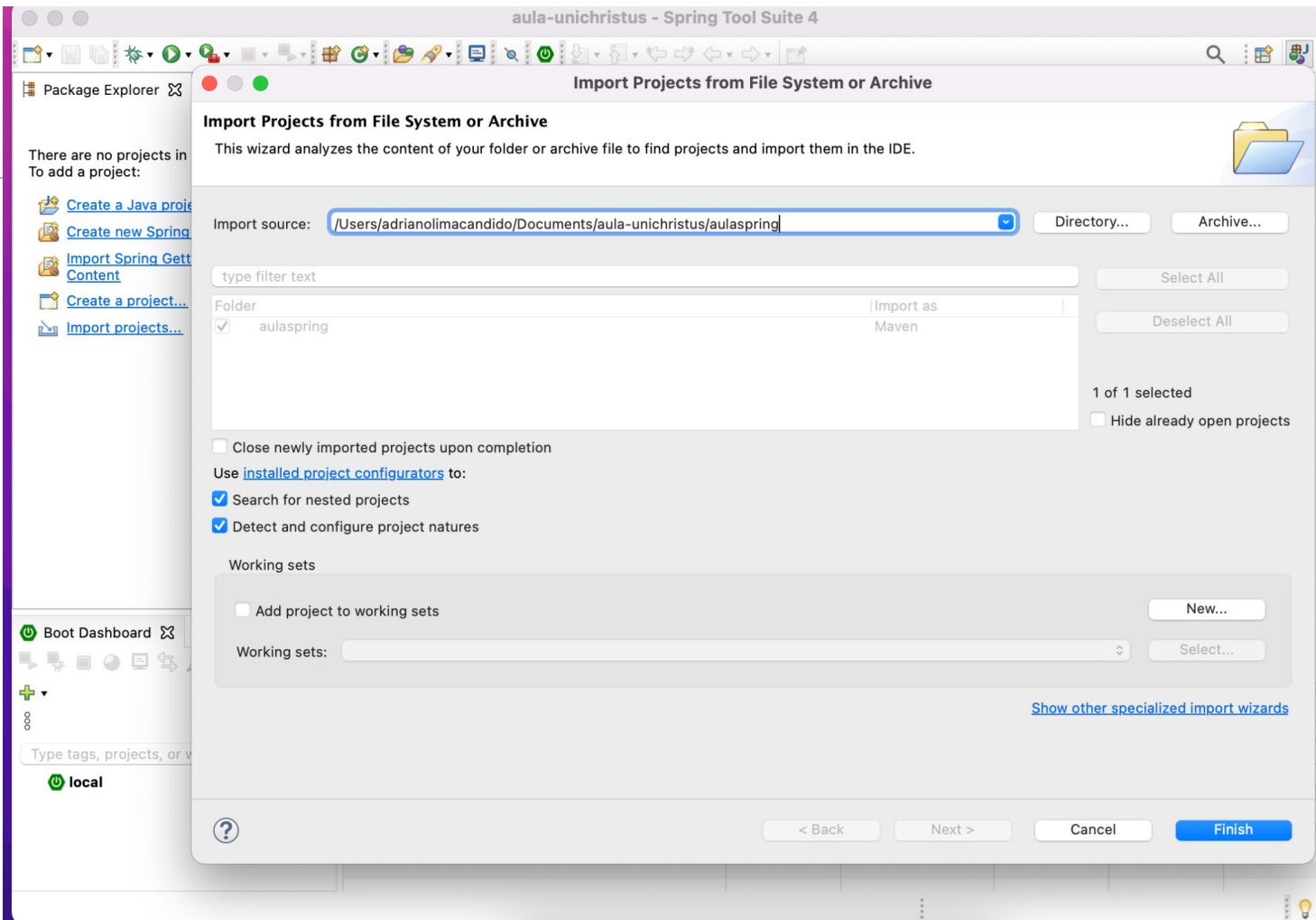
EXPLORE CTRL + SPACE

SHARE...



Spring STS

- Extraí o arquivo o qual foi realizado o download
- Importa no Eclipse (Spring STS)
- Espere a realização dos downloads das dependências
- **Execute como um Java Application**



Crie a classe model da entidade

- User
 - Id
 - Name
 - Price

```
@Entity
```

```
public class Product {
```

```
    @Id
```

```
    @GeneratedValue(strategy = GenerationType.IDENTITY)
```

```
    private Long id;
```

```
    private String name;
```

```
    private Double price;
```

Crie a interface repository da entidade

```
@Repository
```

```
public interface ProductRepository extends
```

```
JpaRepository<Product, Long>{
```

Crie a classe service da entidade

```
@Service  
  
public class ProductService {  
  
    @Autowired  
    private ProductRepository repository;
```

Crie o controller da entidade

```
@RestController  
  
@RequestMapping("/api/product")  
  
public class ProductController {  
    @Autowired  
    private ProductService service;
```


POST - Spring

```
@PostMapping
```

```
public Product create(@RequestBody Product product) {
```

GET ALL - Spring

```
@GetMapping
```

```
public List<Product> findAll() {
```

PostgreSQL

- resources/application.properties
 - Configurações do PostgreSQL
- Criar banco chamado (curso)

PostgreSQL

```
spring.datasource.driver-class-name=org.postgresql.Driver  
spring.datasource.url=jdbc:postgresql://localhost/curso  
spring.datasource.username=postgres  
spring.datasource.password=postgres  
server.port=8080
```

```
spring.jpa.properties.hibernate.dialect = org.hibernate.dialect.PostgreSQL92Dialect  
spring.jpa.hibernate.ddl-auto=update  
spring.jpa.show-sql=false
```

Links importantes

- <https://spring.io/projects>
- <https://start.spring.io/>
- <https://blog.algaworks.com/spring-boot/>
- <https://blog.algaworks.com/como-criar-web-services-restful-com-spring-boot>

Dúvidas??

E-mail: analista.adrianolima@gmail.com

Whats: (88) 9 9729-2878