

VEM SER

Módulo 3 - Testes de API REST REST Assured (Parte 3)



DBC



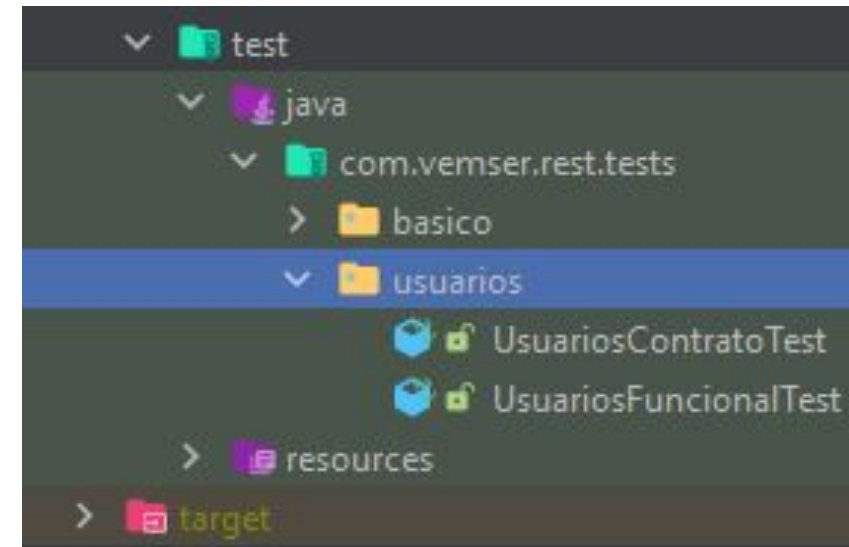
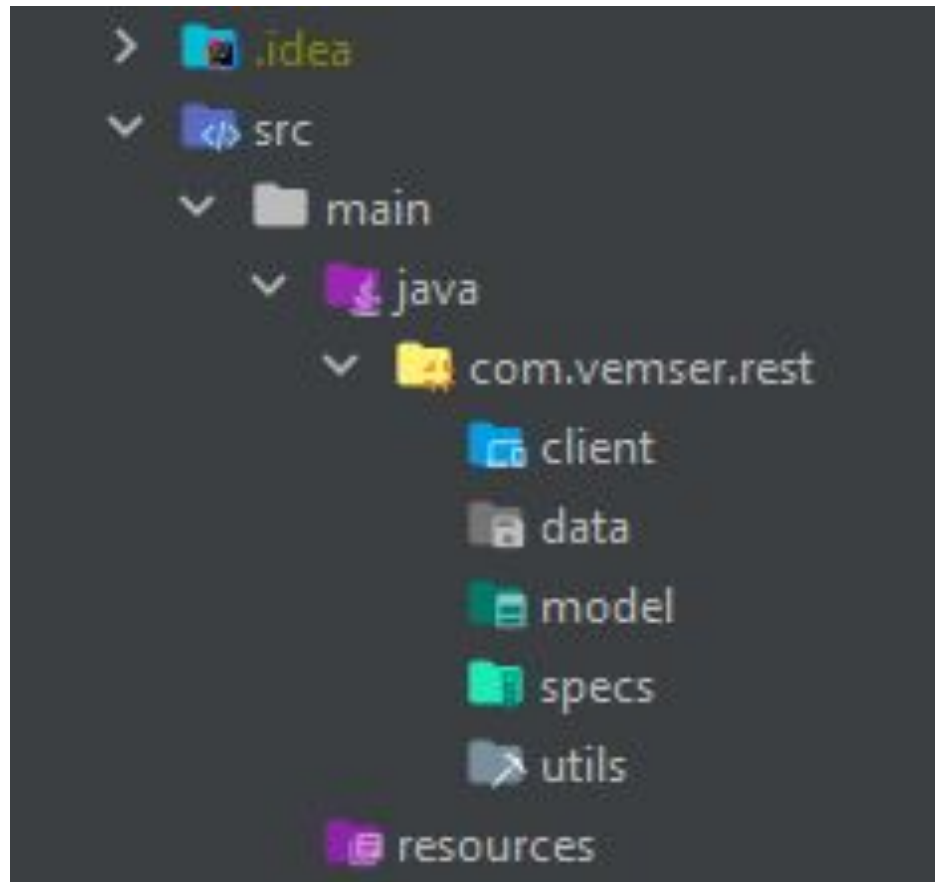
Aula 6 - Rest Assured (Parte 3)

- Montando o Ambiente
- Data Factory
 - Lombok
 - Data Faker
- Client-Test Model
- Spec Builder
- Referências

Montando
o
Ambiente



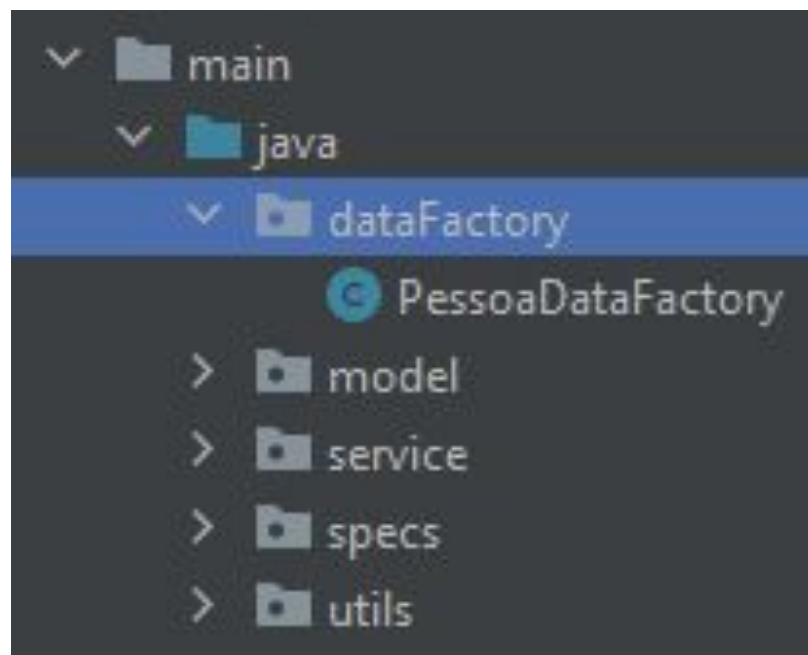
Montando o Ambiente: Estrutura de pastas



Test Data Factory



Test Data Factory



Test Data Factory é uma abordagem cujo objetivo é resolver um dos pontos mais problemáticos da automação, a gestão da massa de dados.

Como o próprio nome diz, trata-se de uma fábrica de dados, ou seja, uma forma fácil e extensível de gerar dados de teste.

Test Data Factory

Como implementar:

1. Criar uma classe para modelar os dados;
2. Criar uma classe de fábrica para consumir os dados;
3. Modificar a classe para gerar dados dinâmicos
4. Usar a classe de fábrica no teste.

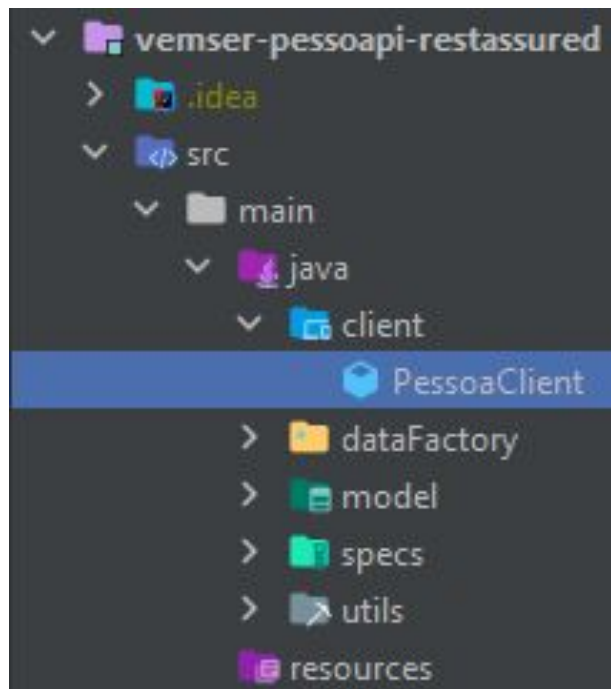
Dependências:

1. **Lombok:** <https://mvnrepository.com/artifact/org.projectlombok/lombok/1.18.24>
2. **Data Faker:** <https://mvnrepository.com/artifact/net.datafaker/datafaker/1.7.0>

Client-Test Model



Client-Test Model



Separação de responsabilidades:

client → conterá toda a configuração para realizar requisições http.

test → se concentrará no próprio teste (ações, asserções)

Spec



Spec

Request e Response Specifications

- **Consistência:** Mantém os testes organizados e consistentes.
- **Reutilização:** Permite a reutilização de código comum em múltiplos testes.

Request Specification:

- Define pré-configurações para os requests.

Response Specification:

- Define Expectativas para as respostas.

Doc: <https://github.com/rest-assured/rest-assured/wiki/Usage#specification-re-use>

Projeto: Credenciais

```
public static String tokenAdmin() {  
    AuthModel auth = new AuthModel();  
    auth.setLogin(Manipulacao.getProp().getProperty("prop.login"));  
    auth.setSenha(Manipulacao.getProp().getProperty("prop.senha"));  
  
    String tokenAdmin =  
        given() RequestSpecification  
            .contentType(ContentType.JSON)  
            .body(auth)  
        .when()  
            .post(baseUrl) Response  
        .then() ValidatableResponse  
            .extract().asString()  
        ;  
    return tokenAdmin;  
}
```

Referências

<http://www.eliasnogueira.com/test-data-factory-why-and-how-to-use/>

<https://testingboss.com/blog/client-test-model-for-rest-api-testing/>

<https://www.ontestautomation.com/using-the-client-test-model-in-rest-assured-net/>

<https://www.baeldung.com/junit5-assertall-vs-multiple-assertions>

<https://junit.org/junit5/docs/5.0.1/api/org/junit/jupiter/api/Assertions.html>

<https://github.com/datafaker-net/datafaker>



Let's *Tech Up Together*