

Processo Seletivo - Desenvolvedor FullStack

Testes:

Para a realização dos testes, primeiramente foi utilizado o JUnit, mas não se obteve resultado por algum problema que não consegui resolver, o problema consiste em não realizar o cadastramento de venda no banco de dados, como pode ser visto no arquivo "ApivendasApplicationTests.java".

Com isso, foi utilizado o Posman para a realização dos testes, para a visualização dos mesmos seguem os passos realizados, acompanhados de seus prints.

1. Cadastro de 4 vendedores:

The image displays two screenshots of the Postman REST client interface, showing the configuration for two separate POST requests to the endpoint `localhost:8080/vendedor`.

Top Screenshot:

- Method:** POST
- URL:** localhost:8080/vendedor
- Body Type:** raw (selected)
- Body Content:**

```
1 {  
2   "nome": "Joao"  
3 }
```

Bottom Screenshot:

- Method:** POST
- URL:** localhost:8080/vendedor
- Body Type:** raw (selected)
- Body Content:**

```
1 {  
2   "nome": "Maria"  
3 }
```

POST localhost:8080/vendedor

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL BETA JSON ▼

```
1 {  
2   "nome": "Pedro"  
3 }
```

POST localhost:8080/vendedor

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL BETA JSON ▼

```
1 {  
2   "nome": "Lucas"  
3 }
```

2. Recuperar todos os vendedores cadastrados:

GET

localhost:8080/vendedor/findAll

Params

Authorization

Headers (9)

Body ●

Pre-request

Query Params

KEY

Key

Body

Cookies

Headers (3)

Test Results

Pretty

Raw

Preview

Visualize BETA

JSON



```
1  [
2    {
3      "id": 1,
4      "nome": "Joao"
5    },
6    {
7      "id": 2,
8      "nome": "Maria"
9    },
10   {
11     "id": 3,
12     "nome": "Pedro"
13   },
14   {
15     "id": 4,
16     "nome": "Lucas"
17   }
18 ]
```

3. Cadastro de 6 vendas: 4 para o vendedor de id = 2 e 2 para o vendedor de id = 3.

POST localhost:8080/venda

Params Authorization Headers (9) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL BETA JSON

```
1 {
2   "dataVenda": "2019-10-10",
3   "valor": 1.99,
4   "vendedor": {
5     "id": 2,
6     "nome": "Maria"
7   }
8 }
```

POST localhost:8080/venda

Params Authorization Headers (1) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL BETA JSON

```
1 {
2   "dataVenda": "2019-10-11",
3   "valor": 3.49,
4   "vendedor": {
5     "id": 2,
6     "nome": "Maria"
7   }
8 }
```

POST localhost:8080/venda

Params Authorization Headers (1) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL BETA JSON

```
1 {
2   "dataVenda": "2019-10-12",
3   "valor": 4.89,
4   "vendedor": {
5     "id": 2,
6     "nome": "Maria"
7   }
8 }
```

POST localhost:8080/venda

Params Authorization Headers (1) Body ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL BETA JSON ▼

```
1 {  
2   "dataVenda": "2019-10-12",  
3   "valor": 5.29,  
4   "vendedor": {  
5     "id": 2,  
6     "nome": "Maria"  
7   }  
8 }
```

POST localhost:8080/venda

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL BETA JSON ▼

```
1 {  
2   "dataVenda": "2019-10-10",  
3   "valor": 1.29,  
4   "vendedor": {  
5     "id": 3,  
6     "nome": "Pedro"  
7   }  
8 }
```

POST localhost:8080/venda

Params Authorization Headers (9) Body ● Pre-request Script Tests Settings

● none ● form-data ● x-www-form-urlencoded ● raw ● binary ● GraphQL BETA JSON ▼

```
1 {  
2   "dataVenda": "2019-10-11",  
3   "valor": 1.89,  
4   "vendedor": {  
5     "id": 3,  
6     "nome": "Pedro"  
7   }  
8 }
```

4. Recuperar lista de vendedores:

GET

localhost:8080/vendedor/2019-10-09/2019-10-13

Params

Authorization

Headers (9)

Body

Pre-request Sc

☒ none

☐ form-data

☐ x-www-form-urlencoded

☐ raw

☐ b

Body

Cookies

Headers (3)

Test Results

Pretty

Raw

Preview

Visualize BETA

JSON



```
1  [
2    [
3      "Maria",
4      4,
5      1
6    ],
7    [
8      "Pedro",
9      2,
10     0
11   ]
12 ]
```