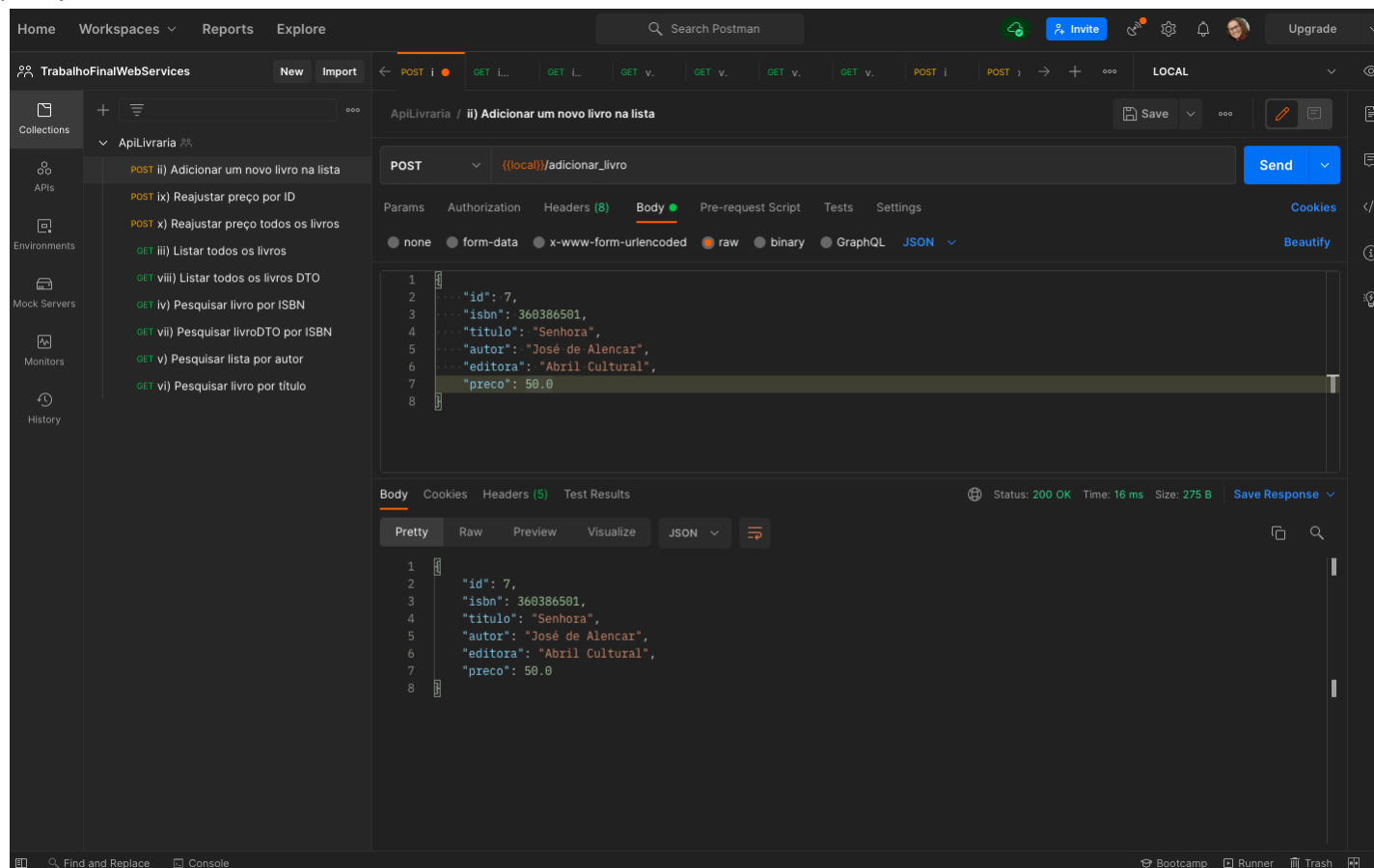




Instituto Federal de Santa Catarina
Curso Técnico em Desenvolvimento de Sistemas
Unidade Curricular Serviços Web
Professor Marcos André Pisching
Estudante **Raquel Darelli Michelon**

Atividade Final: Livraria API

ii) adicionar(Livro livro) → para adicionar um novo livro na lista



iii) getAll() → para retornar todos os livros da lista, cada qual com todo os seus dados (atributos)

The screenshot shows the Postman interface with a GET request to the endpoint `{{local}}/listar_livros`. The request is successful, returning a 200 OK status with a response time of 17 ms and a size of 827 B. The response body is displayed in JSON format, showing an array of three book objects.

KEY	VALUE	DESCRIPTION
Key	Value	Description

```
1 {
2   {
3     "id": 1,
4     "isbn": 987653457,
5     "titulo": "O alienista",
6     "autor": "Machado de Assis",
7     "editora": "Globo",
8     "preco": 20.0
9   },
10  {
11    "id": 2,
12    "isbn": 982364036,
13    "titulo": "Dom Casmurro",
14    "autor": "Machado de Assis",
15    "editora": "LPM",
16    "preco": 12.3
17  },
18  {
19    "id": 3,
20    "isbn": 193874233,
21    "titulo": "Duna",
22    "autor": "Frank Herbert",
23    "editora": "Parábola",
24    "preco": 59.99
25  },
26 }
```

iv) getPorIsbn(int isbn) → para retornar todos os dados de um livro conforme o seu códigoISBN.

The screenshot displays the Postman interface with a workspace named 'TrabalhoFinalWebServices'. The left sidebar shows a collection 'Apilivraria' with several API endpoints. The selected endpoint is 'GET iv) Pesquisar livro por ISBN'. The request is configured with the method 'GET' and the URL 'http://localhost:3000/apilivraria/iv/pesquisar_por_isbn/983742873'. The 'Query Params' section is empty. The response is a 200 OK status with a response time of 14 ms and a size of 275 B. The response body is displayed in JSON format, showing the details of a book.

Request:

```
GET {{local}}/pesquisar_por_isbn/983742873
```

Query Params:

KEY	VALUE	DESCRIPTION
Key	Value	Description

Response:

```
{
  "id": 4,
  "isbn": 983742873,
  "titulo": "Água Viva",
  "autor": "Clarice Lispector",
  "editora": "Editora 34",
  "preco": 39.5
}
```

v) getPorAutor(String autor) → para retornar uma lista de livros dado o nome do autor

The screenshot shows the Postman interface with a workspace named "TrabalhoFinalWebServices". The left sidebar lists a collection "ApiLivraria" with several endpoints. The selected endpoint is "v) Pesquisar lista por autor" (GET v). The request URL is set to "localhost/pesquisar_por_autor/Machado de Assis". The response status is "200 OK" with a time of 7 ms and a size of 378 B. The response body is a JSON array of two book objects.

Request:

- Method: GET
- URL: `localhost/pesquisar_por_autor/Machado de Assis`

Response:

```
1 {
2   {
3     "id": 1,
4     "isbn": 987653457,
5     "titulo": "O alienista",
6     "autor": "Machado de Assis",
7     "editora": "Globo",
8     "preco": 20.0
9   },
10  {
11    "id": 2,
12    "isbn": 982364036,
13    "titulo": "Dom Casmurro",
14    "autor": "Machado de Assis",
15    "editora": "LPM",
16    "preco": 12.3
17  }
18 }
```

vi) getPorTitulo(String titulo) → para retornar um livro conforme o seu título

The screenshot shows the Postman interface with a workspace named 'TrabalhoFinalWebServices'. The left sidebar lists a collection 'ApiLivraria' with several endpoints. The selected endpoint is 'GET vi) Pesquisar livro por título'. The URL bar shows the request: `GET {{local}}/pesquisar_por_titulo/Duna`. The 'Params' tab is active, showing a table for query parameters.

KEY	VALUE	DESCRIPTION
Key	Value	Description

The 'Body' tab is also active, showing the response in JSON format:

```
1 {
2   "id": 3,
3   "isbn": 193874233,
4   "titulo": "Duna",
5   "autor": "Frank Herbert",
6   "editora": "Parábola",
7   "preco": 59.99
8 }
```

At the bottom, the status bar indicates 'Status: 200 OK', 'Time: 10 ms', and 'Size: 265 B'.

vii) getPorIsbnDTO(int isbn) → retorna um objeto DTO contendo título, autor, editora e preço.

The screenshot shows the Postman interface with a workspace named "TrabalhoFinalWebServices". The left sidebar lists collections, with "ApiLivraria" expanded. The main panel shows a GET request to the endpoint `{{local}}/pesquisar_por_isbnDTO/987653457`. The request is saved and has a status of 200 OK. The response body is displayed in JSON format, showing the following data:

KEY	VALUE	DESCRIPTION
Key	Value	Description

```
1  {
2    "titulo": "O alienista",
3    "autor": "Machado de Assis",
4    "editora": "Globo",
5    "preco": 20.0
6  }
```

The status bar at the bottom indicates the request was successful (200 OK) with a response time of 10 ms and a size of 246 B.

viii) getAlIDTO() → retorna os atributos título, autor, editora e preço de todos os livros.

The screenshot shows the Postman interface with a REST client request configured for the endpoint `{{local}}/livrosDTO` using the GET method. The request is part of a collection named 'ApiLivraria' under the workspace 'TrabalhoFinalWebServices'. The response is a 200 OK status with a response time of 15 ms and a size of 595 B. The response body is displayed in JSON format, showing an array of four book objects.

KEY	VALUE	DESCRIPTION
Key	Value	Description

```
1 {
2   {
3     "titulo": "O alienista",
4     "autor": "Machado de Assis",
5     "editora": "Globo",
6     "preco": 20.0
7   },
8   {
9     "titulo": "Dom Casmurro",
10    "autor": "Machado de Assis",
11    "editora": "LPM",
12    "preco": 12.3
13  },
14  {
15    "titulo": "Duna",
16    "autor": "Frank Herbert",
17    "editora": "Parábola",
18    "preco": 59.99
19  },
20  {
21    "titulo": "Água Viva",
22    "autor": "Clarice Lispector",
23    "editora": "Editora 34",
24    "preco": 39.5
25  }
26 }
```

ix) reajustarPrecoPorId(int id, int taxa) → método para reajustar o preço de um livro da lista conforme o seu id. O reajuste será feito de acordo com a taxa recebida no parâmetro.

The screenshot displays the Postman interface with a REST client request and its response.

Request Details:

- Collection:** TrabalhoFinalWebServices
- API:** ApiLivraria
- Endpoint:** ix) Reajustar preço por ID
- Method:** POST
- URL:** {{local}}/reajustar_preco
- Body:**

```
{
  "id": 1,
  "taxa": 10
}
```
- Headers (8):** none, form-data, x-www-form-urlencoded, raw, binary, GraphQL, JSON
- Status:** 200 OK
- Time:** 12 ms
- Size:** 270 B

Response Details:

- Body:**

```
{
  "id": 1,
  "isbn": 987653457,
  "titulo": "O alienista",
  "autor": "Machado de Assis",
  "editora": "Globo",
  "preco": 22.0
}
```


x) reajustarPreco(int taxa) → reajusta o preço de todos os livros conforme a taxa informada por parâmetro.

The screenshot shows the Postman interface with a REST client request configured. The left sidebar displays the 'Collections' panel with a collection named 'TrabalhoFinalWebServices'. Under this collection, there is a sub-collection 'ApiLivraria' containing several API endpoints. The selected endpoint is 'POST x) Reajustar preço todos os livros'. The main panel shows the request details for this endpoint, including the URL, method, and body.

Request Details:

- Method:** POST
- URL:** `{{local}}/reajustar_precos`
- Body:** The body is set to 'raw' (JSON) and contains the following JSON payload:

```
1 {
2   ... "taxa": 10
3 }
```
- Response:** The response is displayed in the 'Body' tab, showing a JSON array of book objects. The status is 200 OK, time is 27 ms, and size is 844 B.

Response Body (JSON):

```
1 {
2   {
3     "id": 1,
4     "isbn": 987653457,
5     "titulo": "O alienista",
6     "autor": "Machado de Assis",
7     "editora": "Globo",
8     "preco": 24.2
9   },
10  {
11    "id": 2,
12    "isbn": 982364036,
13    "titulo": "Dom Casmurro",
14    "autor": "Machado de Assis",
15    "editora": "LPM",
16    "preco": 13.530000000000001
17  },
18  {
19    "id": 3,
20    "isbn": 193874233,
21    "titulo": "Duna",
22    "autor": "Frank Herbert",
23    "editora": "Parábola",
24  }
25 }
```