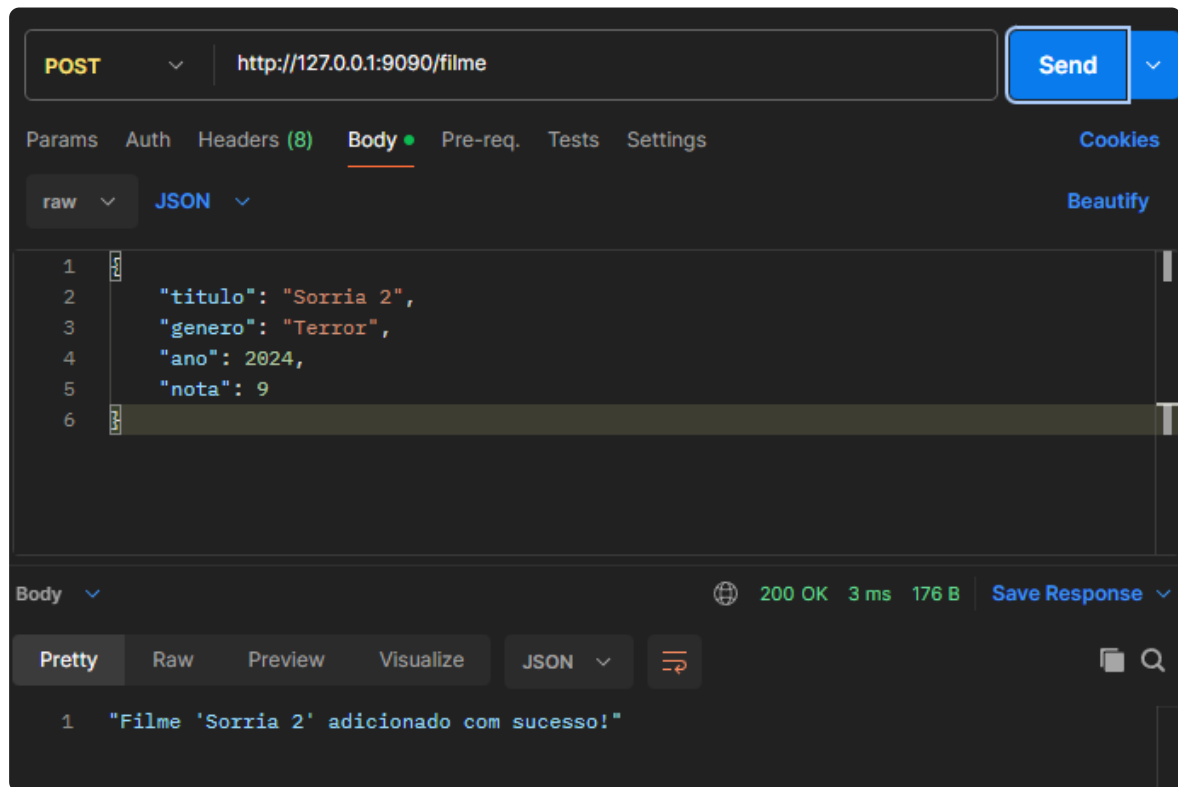


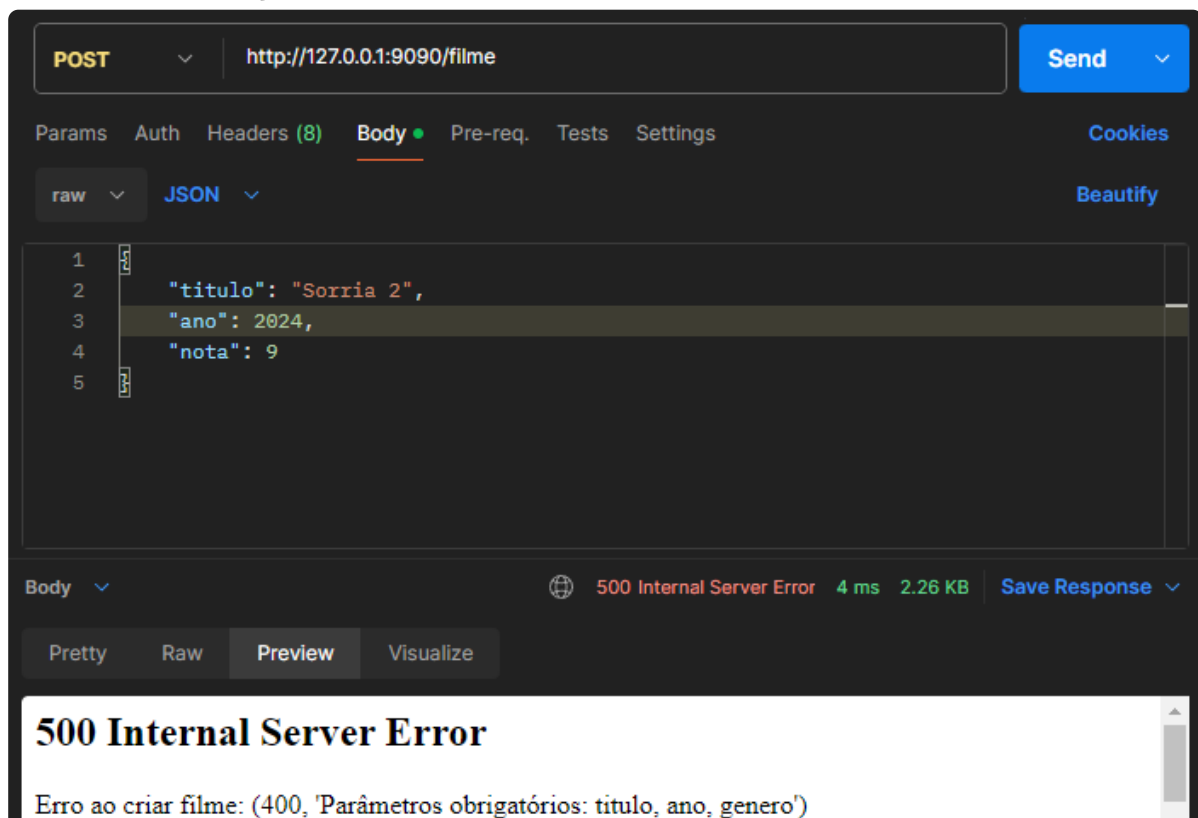
Alunos: Gabriel de O. e S. P. Barreto RA: 2326221, Johanna Kirchner RA: 2265010

Neste trabalho, continuamos desenvolvendo o 'Gerenciador de Filmes'.

Para a função `criar_filme`, a requisição precisa conter os campos `titulo`, `ano` e `genero`, sendo que o campo `nota` é opcional.



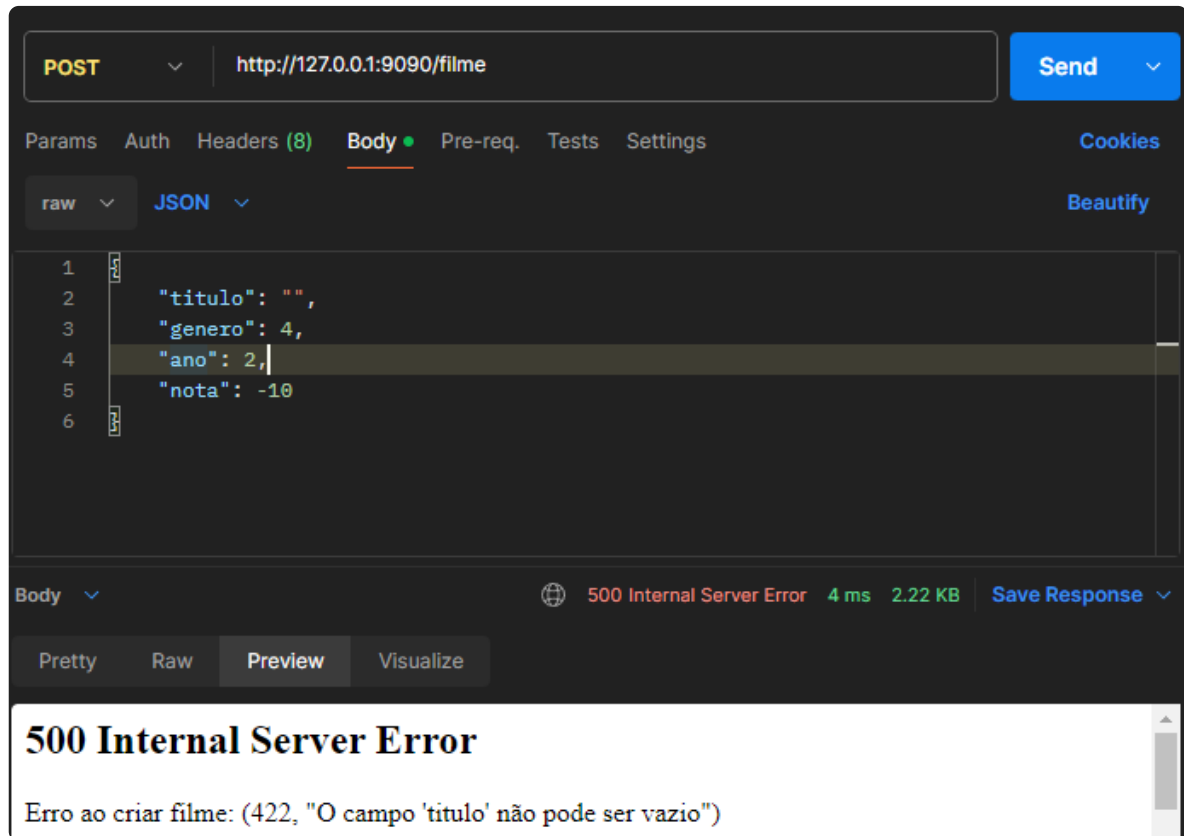
Quando um desses campos obrigatórios esteja ausente, é gerado um erro com código HTTP 400, que indica que a requisição está mal formada.:



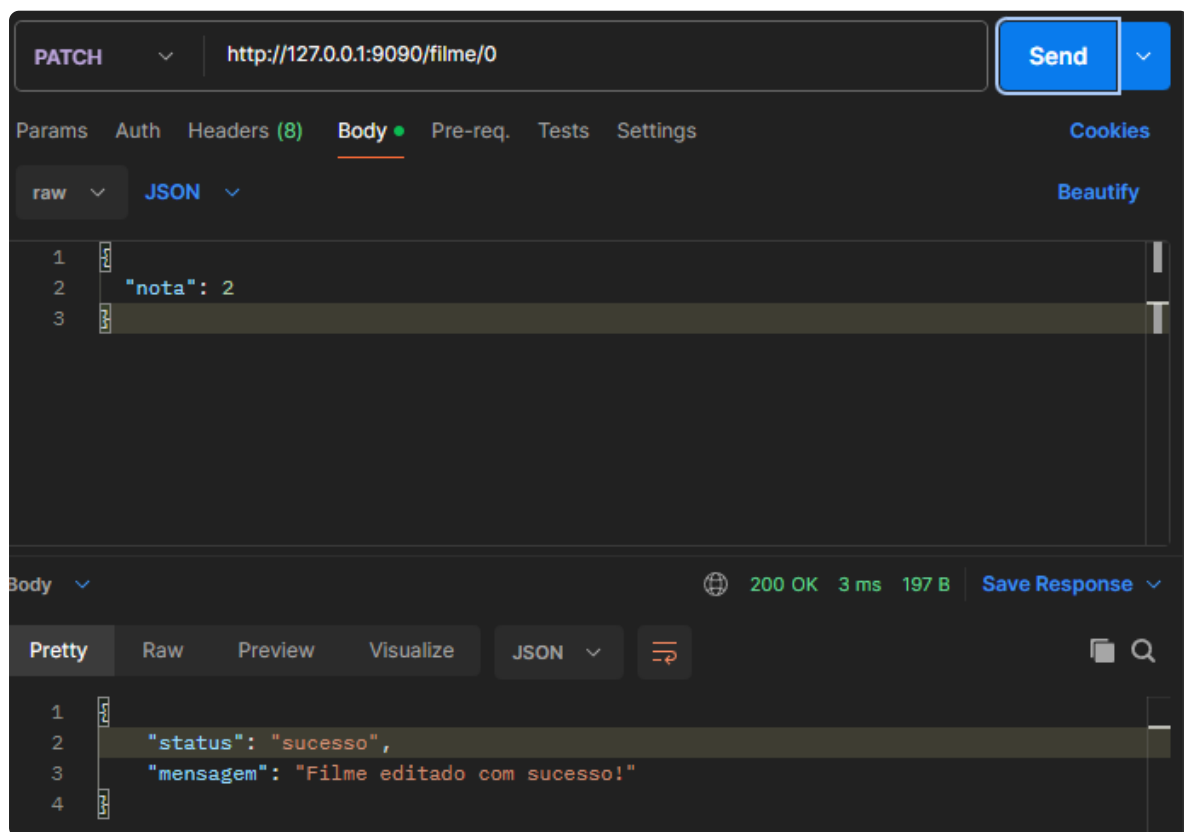
As validações para cada campo são:

- `titulo` e `genero` são campos de texto, então é validado se não estão vazios;
- `ano` valida se está entre 1900 e 2100;
- `nota` é opcional, mas caso presente, deve ser um número maior que zero.

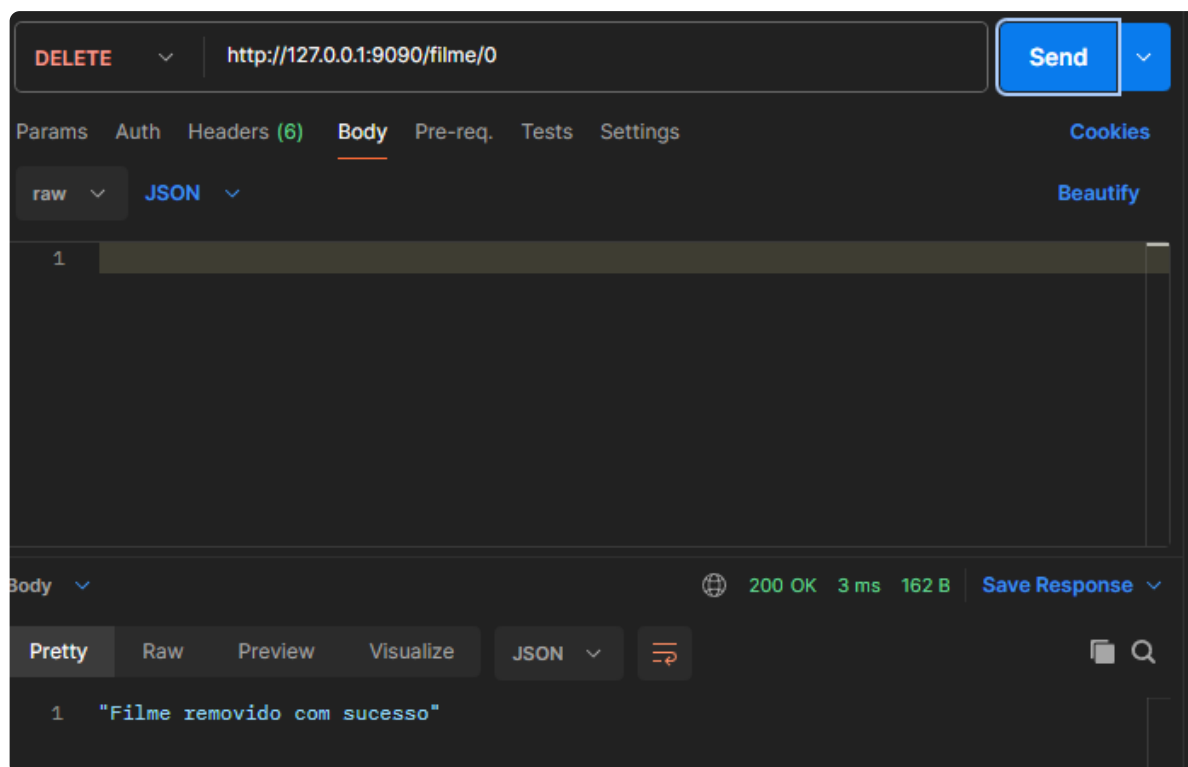
Quando uma dessas validações não são atendidas, é retornado um erro com código HTTP 422, que indica que a sintaxe da requisição está correta mas não foi possível processar os conteúdos internos.



Para `editar_filme`, a requisição necessita de um ID e no body a nota que será atribuída, com ambos sendo obrigatórios e também gera erro HTTP 400. Novamente, o campo `nota` tem uma validação de ser número e maior ou igual a zero.



Para `remover_filme`, a requisição necessita apenas do ID do filme a ser removido.



Para `listar_filmes` não é feita nenhuma verificação, pois apenas retorna a lista dos filmes cadastrados.

GET

http://127.0.0.1:9091/filme

Send

Params

Auth

Headers (6)

Body

Pre-req.

Tests

Settings

Cookies

Query Params

	Key	Value	Bulk Edit
	Key	Value	

Body

200 OK3 ms470 BSave Response

Pretty

Raw

Preview

Visualize

JSON

```
1  []
2  {
3    "id": 0,
4    "titulo": "Nome do Filme",
5    "genero": "Ação",
6    "ano": 2024,
7    "nota": 2.0
8  },
9  {
10   "id": 1,
11   "titulo": "Sorria 2",
12   "genero": "Terror",
13   "ano": 2024,
14   "nota": 2.0
15 },
16 {
17   "id": 2,
18   "titulo": "Pânico",
19   "genero": "Terror",
20   "ano": 1996,
21   "nota": 2.0
22 },
23 {
24   "id": 3,
25   "titulo": "Fale Comigo",
26   "genero": "Terror",
27   "ano": 2022,
28   "nota": 5.0
29 }
30 []
```

Por fim, podemos utilizar o recurso de pesquisar por gênero do filme, onde deve ser passado o gênero:

GET

http://127.0.0.1:9091/filme/ação

Send

Params

Auth

Headers (6)

Body

Pre-req.

Tests

Settings

Cookies

Query Params

	Key	Value	Bulk Edit
	Key	Value	

Body

200 OK3 ms226 B

Save Response

Pretty

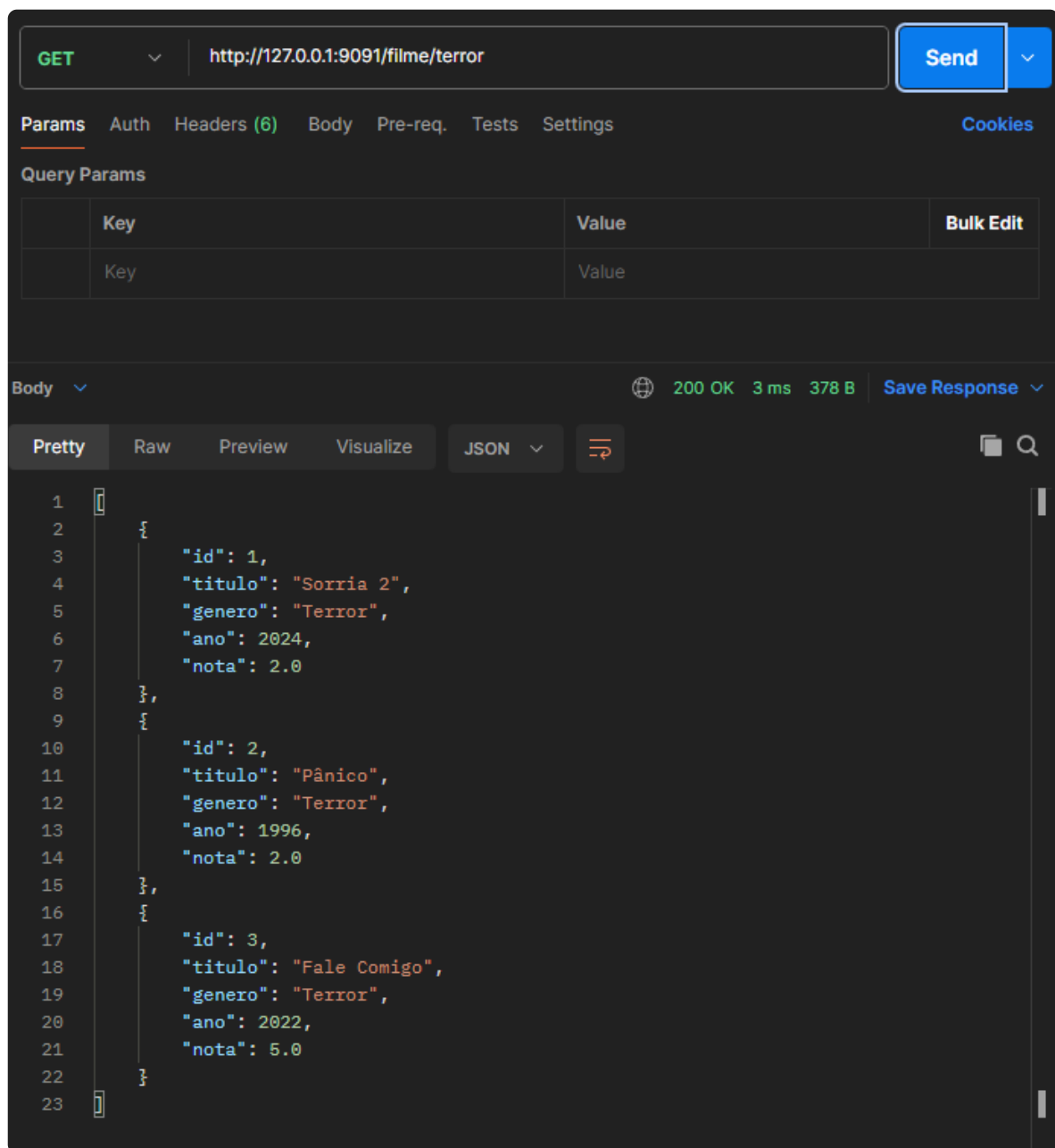
Raw

Preview

Visualize

JSON

```
1  {}
2    {
3      "id": 0,
4      "titulo": "Nome do Filme",
5      "genero": "Ação",
6      "ano": 2024,
7      "nota": 2.0
8    }
9  }
```



## Adicionar

POST http://127.0.0.1:9090/filme

```
{
  "titulo": "Nome do Filme",
  "genero": "Gênero do Filme",
  "ano": 2024,
  "nota": 8.5
}
```

## Editar Nota

PATCH http://127.0.0.1:9090/filme/{id}

```
{
```

```
"nota": 0  
}
```

## Deletar

```
DELETE http://127.0.0.1:9090/filme/{id}
```

## Listar

```
GET http://127.0.0.1:9090/filme
```

## Listar por gênero

```
GET http://127.0.0.1:9090/filme/{gênero}
```