

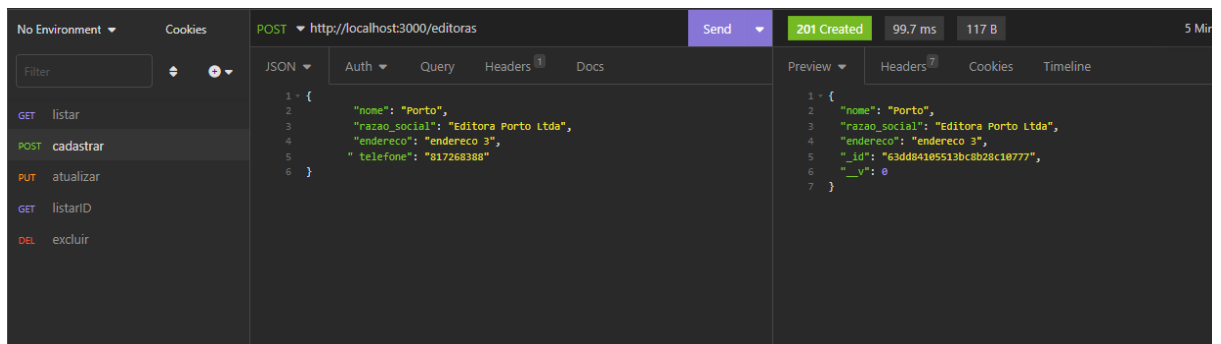
# BACKEND - LIVRARIA

## 1. Criação do Schema

```
JS Editora.js X
Node > apirest > src > models > JS Editora.js > editoraSchema
1 import mongoose from "mongoose";
2
3
4 const editoraSchema = new mongoose.Schema(
5   {
6     nome: {type: String},
7     razao_social: {type: String},
8     endereco: {type: String},
9     telefone: {type: Number}
10  }
11 );
12
13
14 const editoras = mongoose.model('editoras', editoraSchema);
15
16 export default editoras
```

## 2. Realizando Teste Crud com Insomnia

### CADASTRAR



## LISTAR

Insomnia interface showing a GET request to `http://localhost:3000/editoras`. The response is 200 OK, 99 ms, 486 B. The response body is a JSON array of publisher objects:

```
1 + [
2   {
3     "id": "63dd83b65513bc8b28c10773",
4     "nome": "Nova Fronteira",
5     "razao_social": "Editora Nova Fronteira Ltda",
6     "endereco": "endereco 1",
7     "_v": 0
8   },
9   {
10    "id": "63dd83f25513bc8b28c10775",
11    "nome": "Zamar",
12    "razao_social": "Editora Zamar",
13    "endereco": "endereco 2",
14    "_v": 0
15  },
16  {
17    "id": "63dd84e5513bc8b28c10777",
18    "nome": "Porto",
19    "razao_social": "Editora Porto Ltda",
20    "endereco": "endereco 3",
21    "_v": 0
22  },
23  {
24    "id": "63dd85b5513bc8b28c10779",
25    "nome": "Porto",
26    "razao_social": "Editora Porto Ltda",
27    "endereco": "endereco 4",
28    "_v": 0
29  }
30 ]
```

## EXCLUIR

Insomnia interface showing a DELETE request to `http://localhost:3000/editoras/63dd85b5513bc8b28c10779`. The response is 200 OK, 104 ms, 43 B. The response body is a JSON object with a success message:

```
1 + {
2   "message": "Editora removida com sucesso!"
3 }
```

## ATUALIZAR

Insomnia interface showing a PUT request to `http://localhost:3000/editoras/63dd83b65513bc8b28c10773`. The response is 200 OK, 104 ms, 45 B. The response body is a JSON object with a success message:

```
1 + {
2   "message": "Editora atualizada com sucesso!"
3 }
```

### 3. Relacionando Livros e Editora

## ASSOCIAÇÃO

```
Livro.js x
Node > apirest > src > models > Livro.js > livrosSchema
1  import mongoose from "mongoose";
2
3
4  const livrosSchema = new mongoose.Schema(
5    {
6      titulo: {type: String, required: true},
7      Autor: {type: mongoose.Schema.Types.ObjectId, ref: 'autores', required: true},
8      Editora: {type: mongoose.Schema.Types.ObjectId, ref: 'editoras', required: true},
9      numeroPagina: {type: Number}
10   }
11 )
12
13 const livros = mongoose.model('livros', livrosSchema)
14
15 export default livros
```

## BANCO DE DADOS

```
QUERY RESULTS: 1-2 OF 2

  _id: ObjectId('63dd8955513bc8b28c1077f')
  titulo: "1984"
  Autor: ObjectId('63dd80a6434ceea5ddfb5a1e')
  Editora: ObjectId('63dd83b65513bc8b28c10773')
  numeroPagina: 354
  __v: 0

  _id: ObjectId('63dd8a525513bc8b28c10784')
```