**SavvyFix**

A Savvyfix é uma aplicação desenvolvida para entender melhor os padrões de compra dos usuários e, assim, fornecer insights valiosos para otimização de preços. Ao analisar dados de compras, nosso sistema identifica tendências e preferências, permitindo uma precificação mais precisa e personalizada. Isso não só melhora a experiência do usuário, mas também maximiza a receita potencial para os negócios.

Testar os endpoints da Savvyfix é crucial para garantir que o sistema funcione de maneira eficiente e sem erros. Esses testes verificam a integridade e a performance das interfaces de programação, assegurando que os dados de compras sejam analisados corretamente e que as recomendações de precificação sejam precisas e confiáveis.

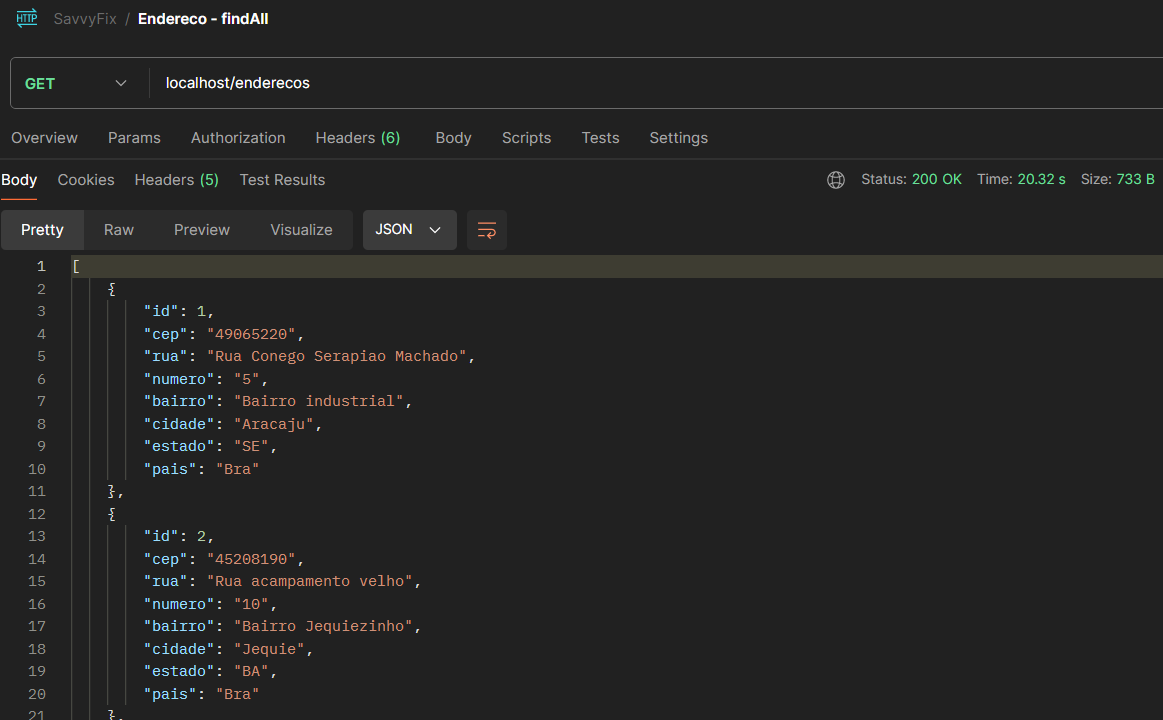
**Teste dos Endpoints**

Para realizar os teste utilizaremos o Postman e junto com os endpoints vamos testar os métodos HTTPS de cada classe para verificar a persistência e recuperação dos dados (GET e POST).

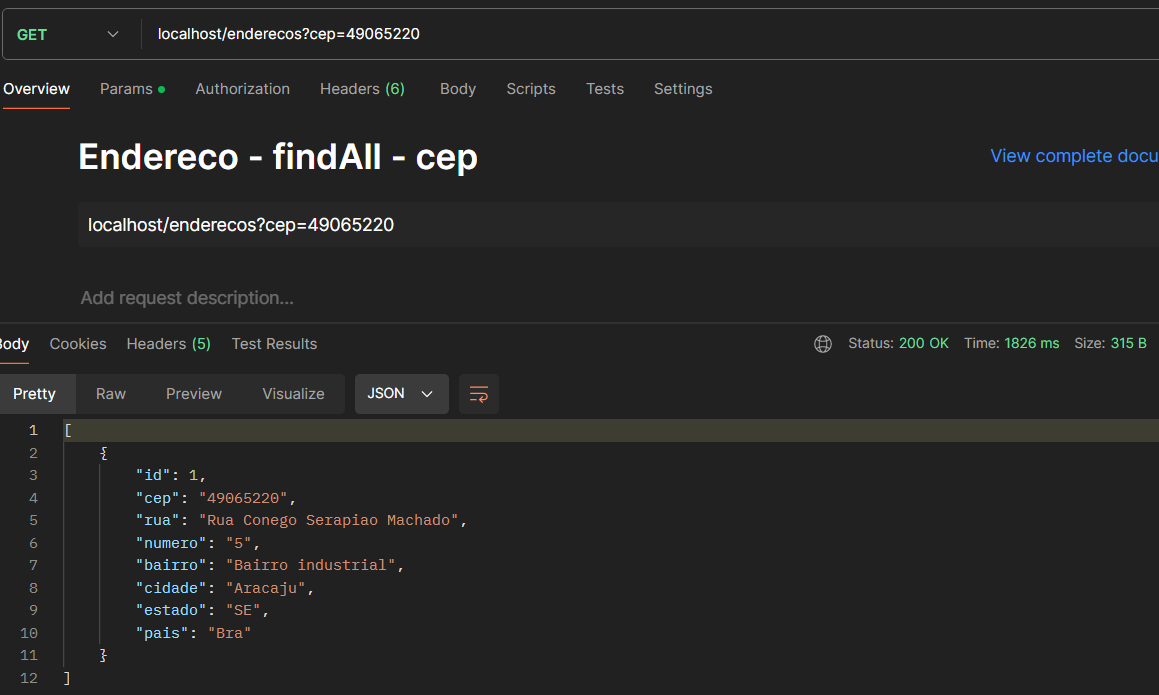
Link do Postman: <https://www.postman.com/luizffarias/workspace/savvyfix/collection/34285252-5e0e6098-c064-49f1-bf47-30c7d7ff030b?action=share&creator=34285252>

**Endereços – GET**

**Endpoint:** localhost/enderecos



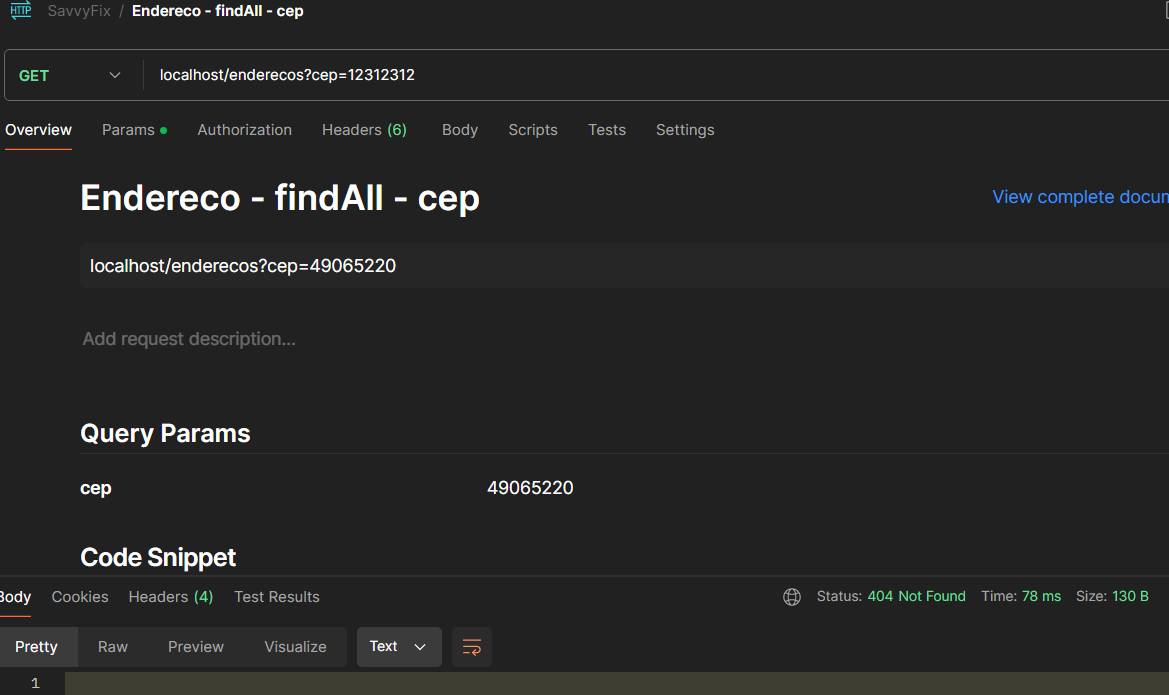
**Endpoint:** localhost/enderecos?cep= 49065220

****

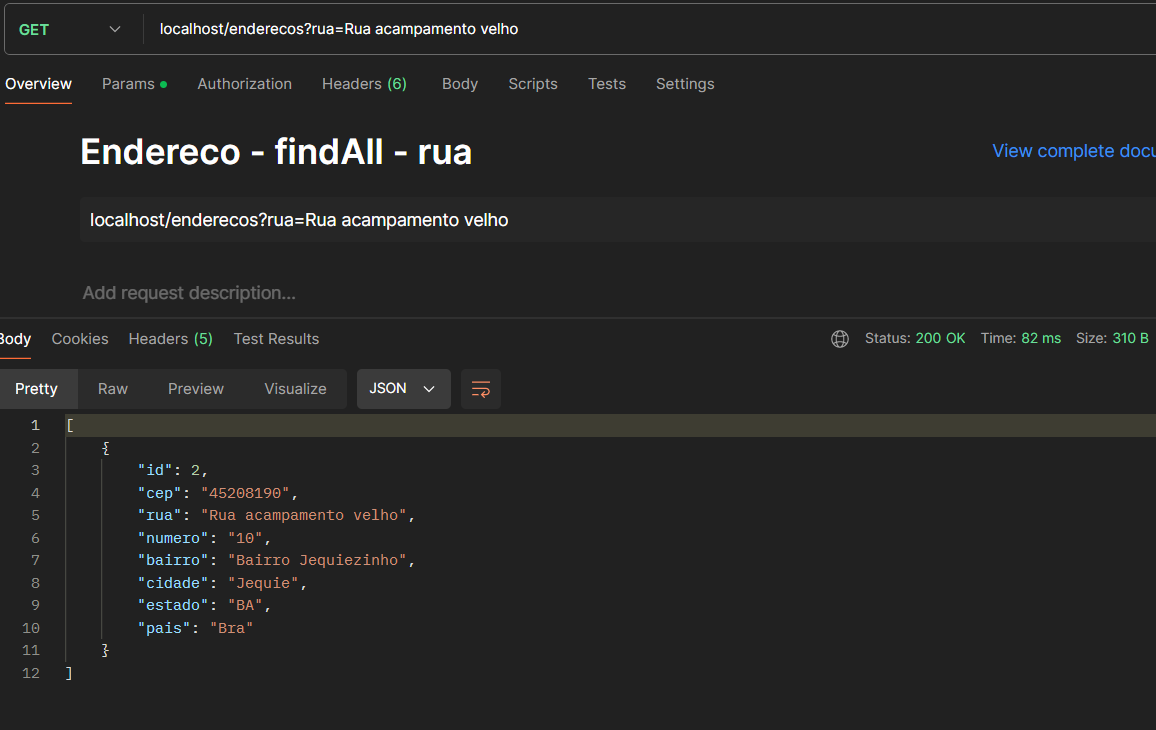
**Endpoint:** localhost/enderecos?cep=12313212

Nessa situação de buscar um endereço por um cep que não existe, a resposta será 404.

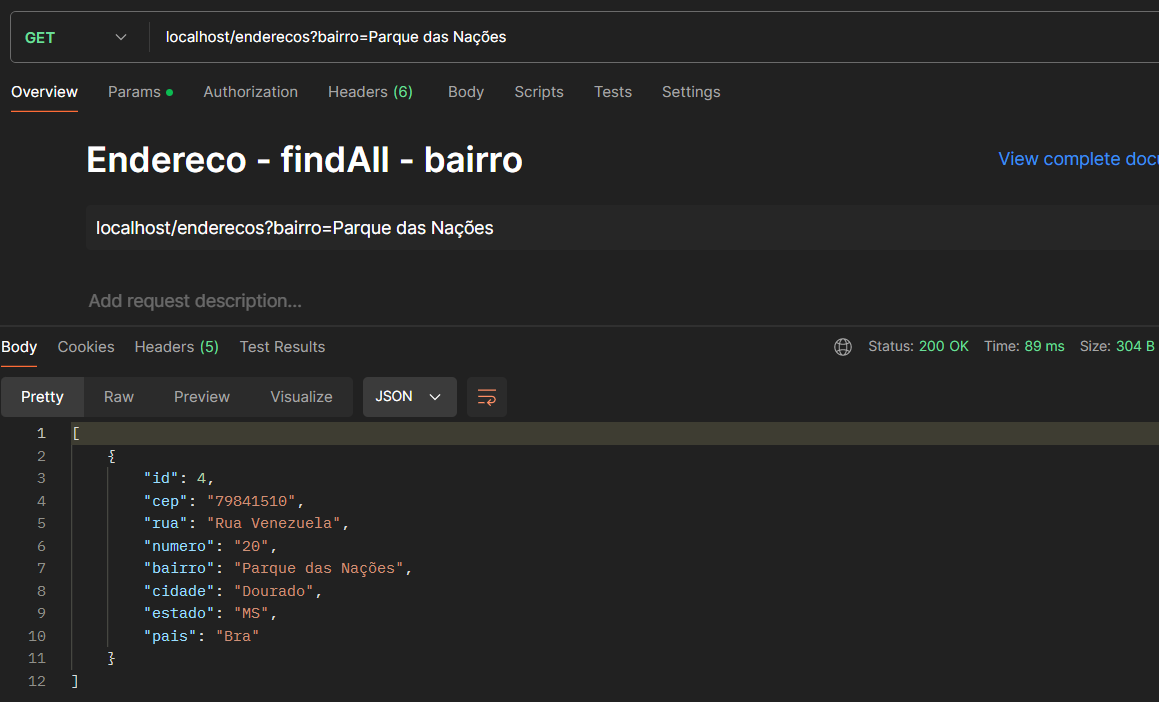
O mesmo ocorre para todos os endpoints de GET por findAll(Example)

****

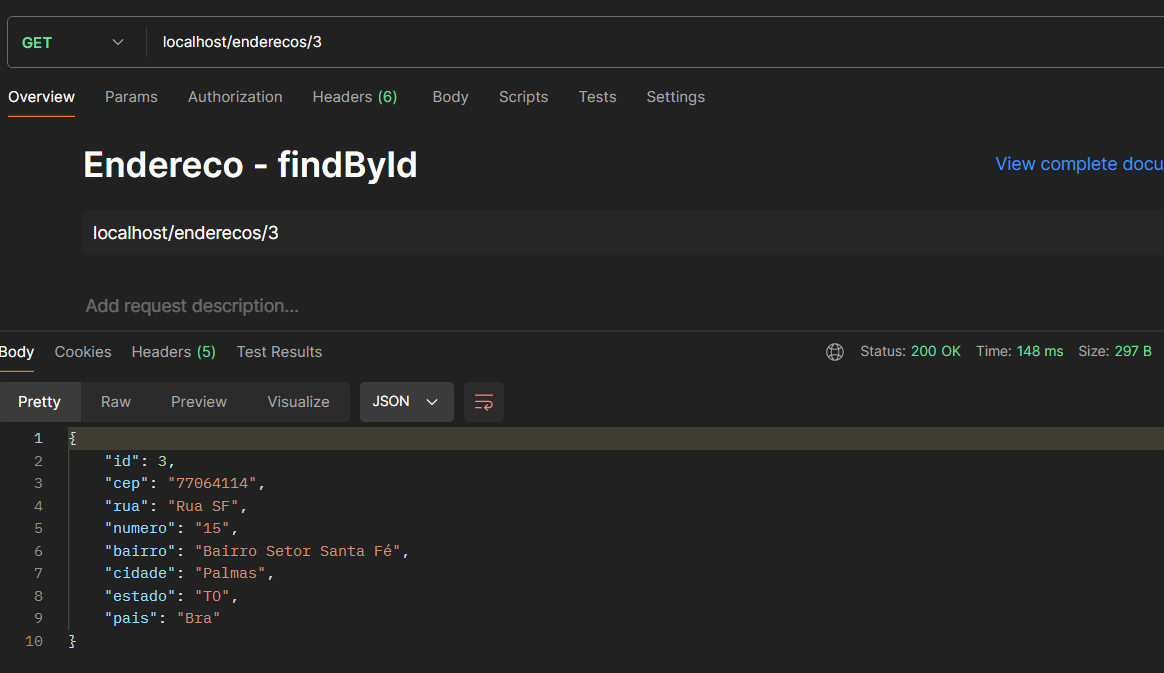
**Endpoint:** localhost/enderecos?rua=Rua acampamento velho



**Endpoint:** localhost/enderecos ?bairro=Parque das Nações

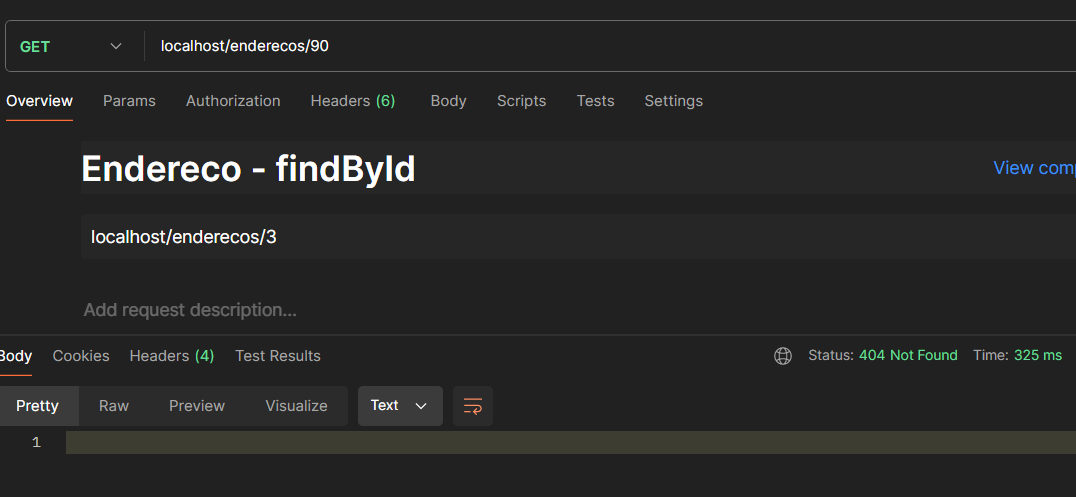
****

**Endpoint:** localhost/enderecos/3



**Endpoint:** localhost/enderecos/90

Ao colocar um id que não existe ele retorna 404.

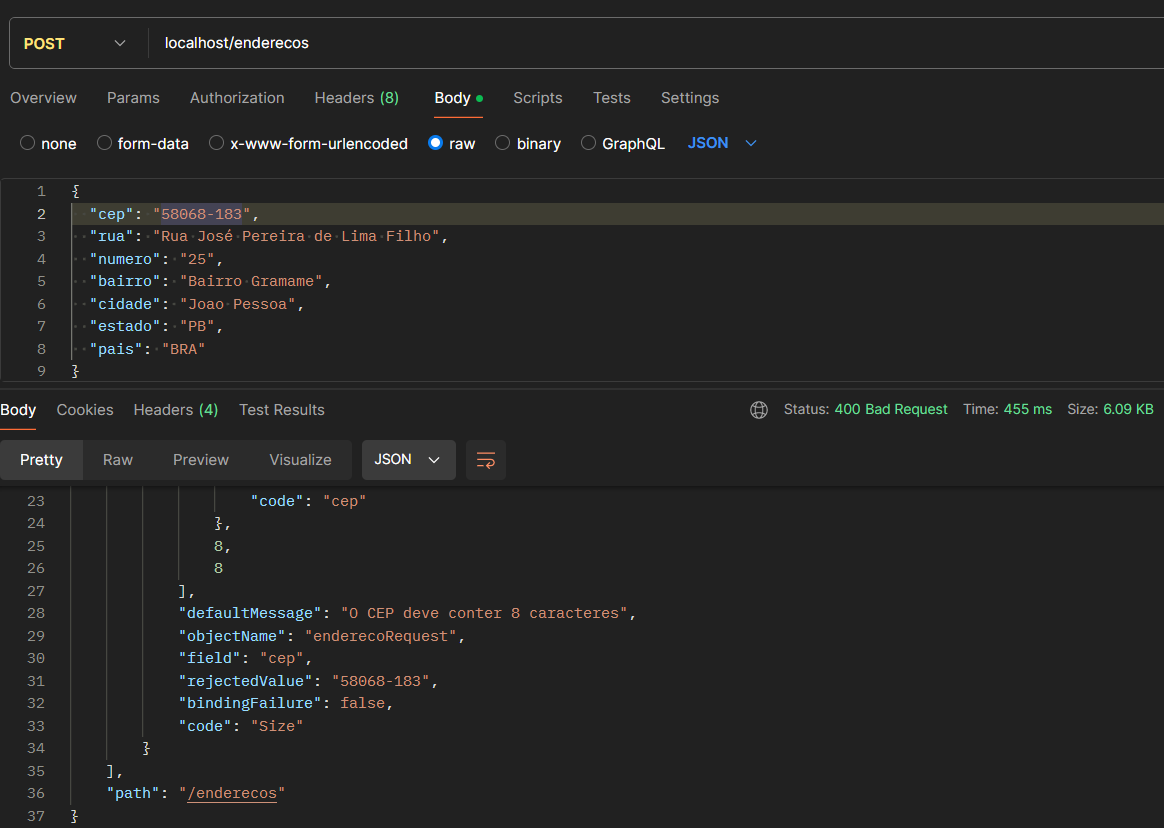


**Endereços – POST**

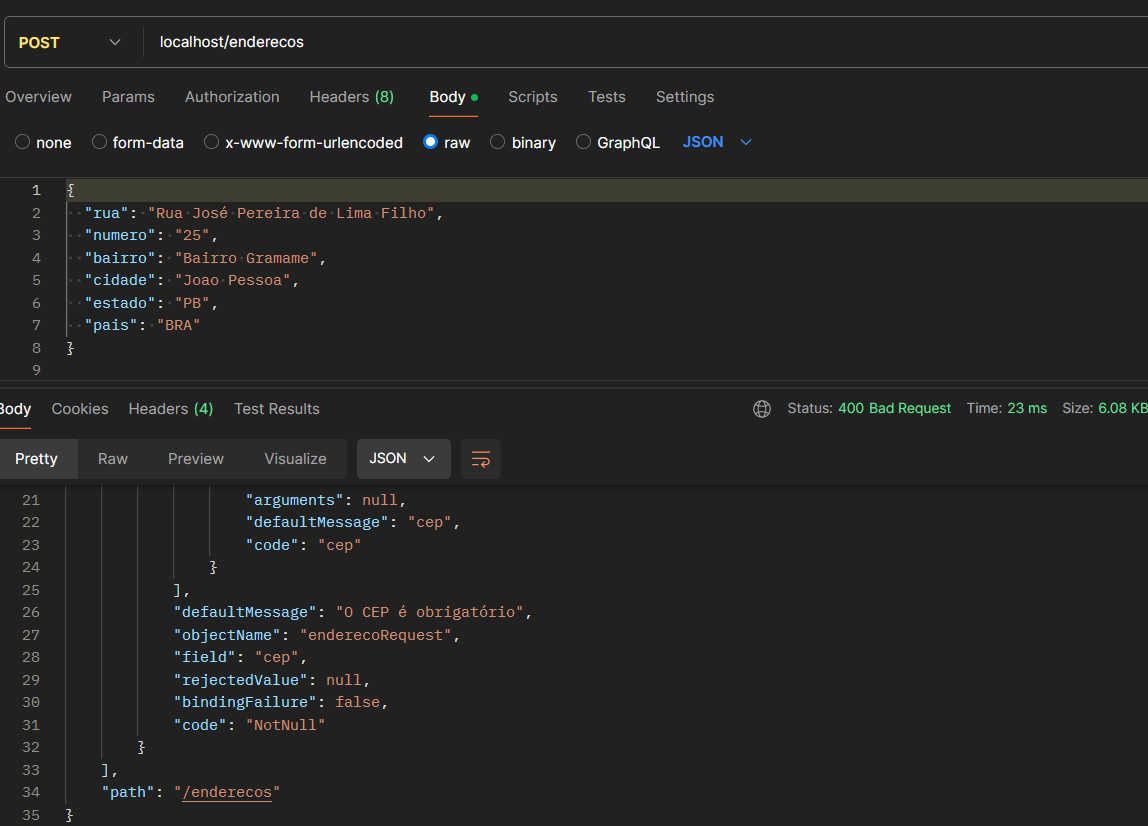
**Endpoint:** localhost/enderecos

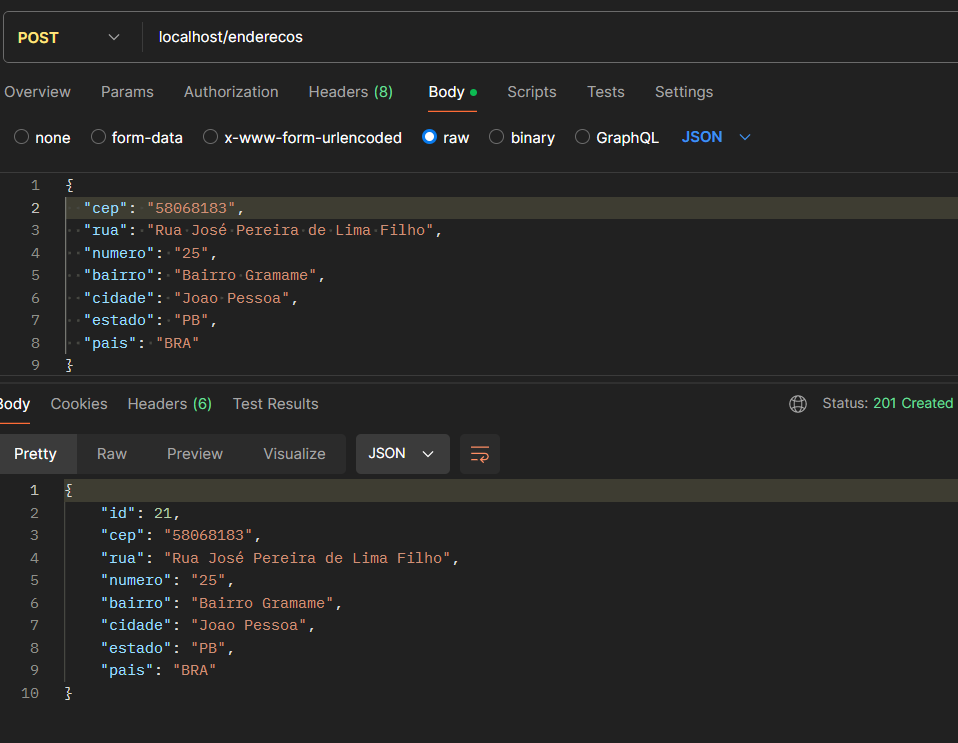
Nessa situação verificamos que para salvar é necessário:

1 – O CEP estar no formato de 8 números sem caracteres especiais.



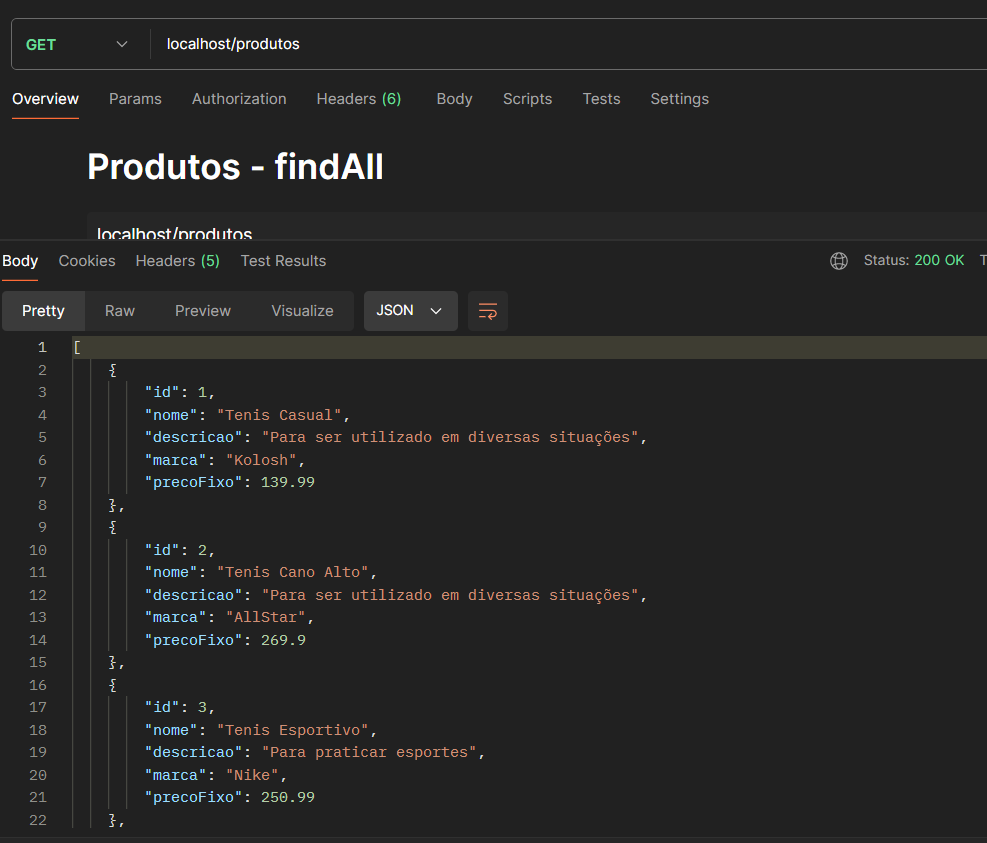
2 - O CEP é obrigatório para salvar os dados, no final do projeto será necessário que o usuário informe apenas o CEP para buscar de forma automática os dados necessários.



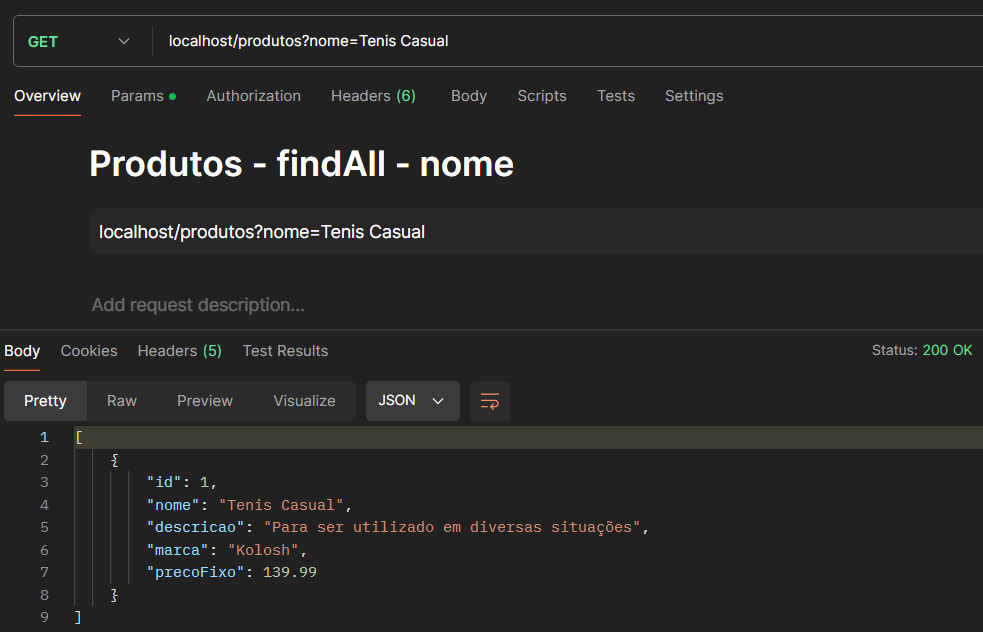


**Produtos – GET**

**Endpoint:** localhost/produtos

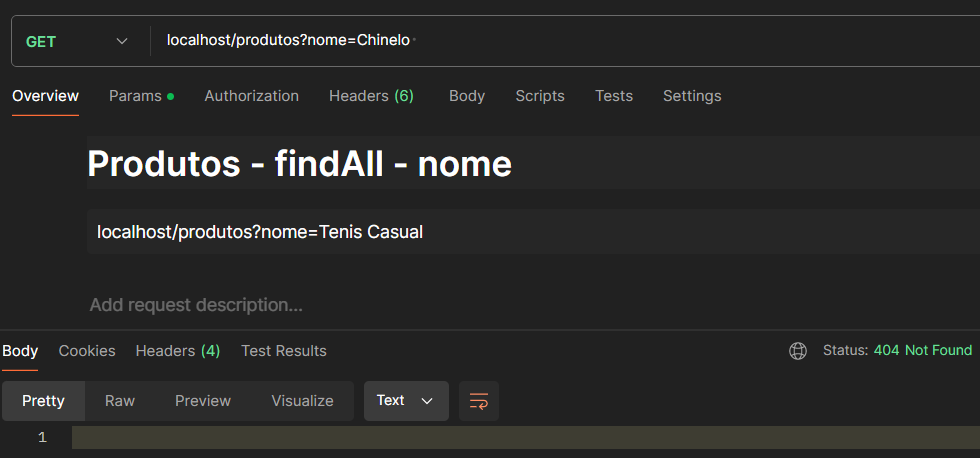
****

**Endpoint:** localhost/produtos?nome=Tenis Casual

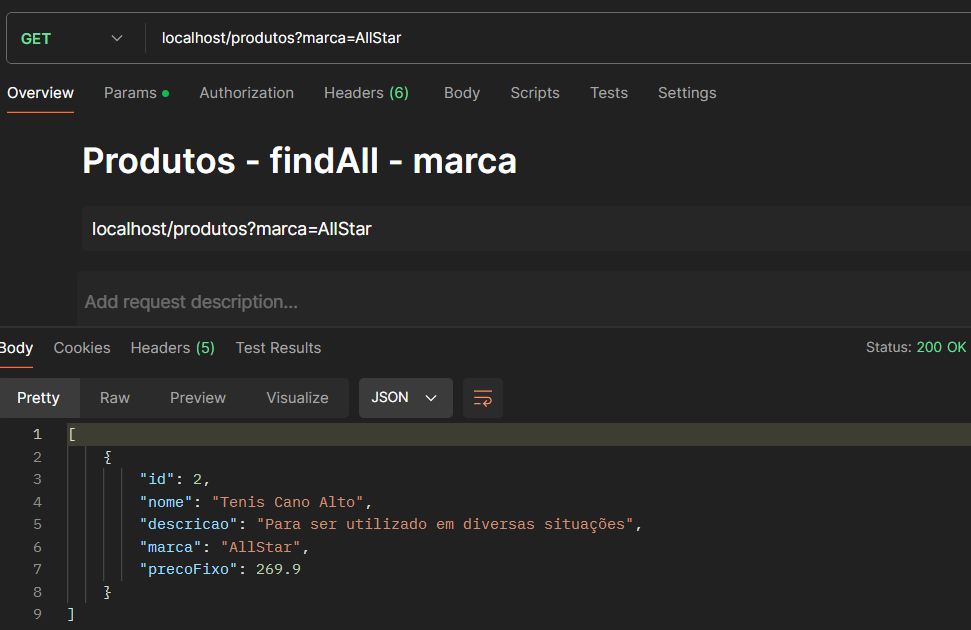


**Endpoint:** localhost/produtos?nome=Chinelo

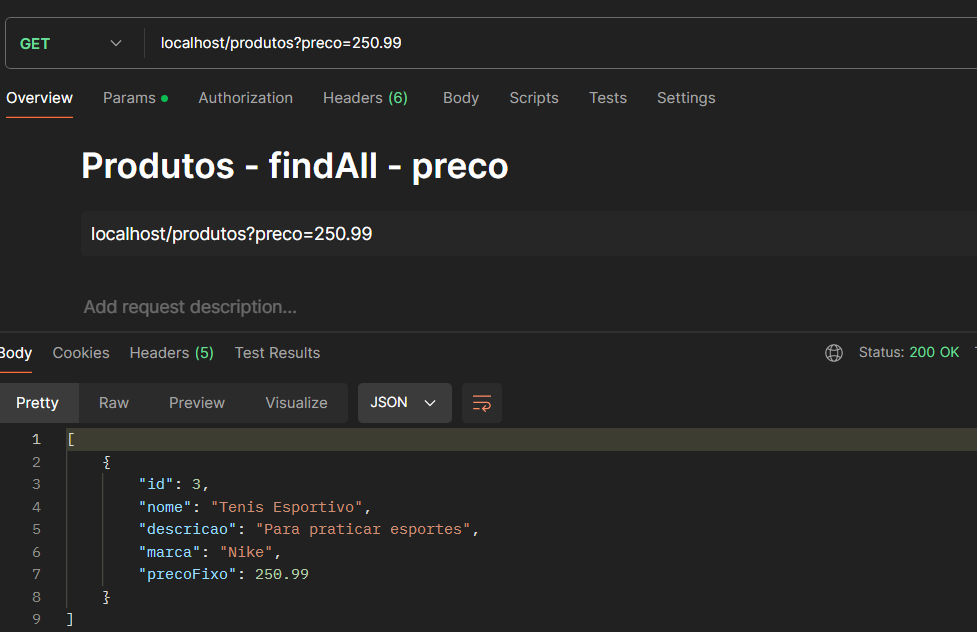
Caso o nome não exista ele retorna um 404. O mesmo ocorre para todos os endpoints de GET por findAll(Example)



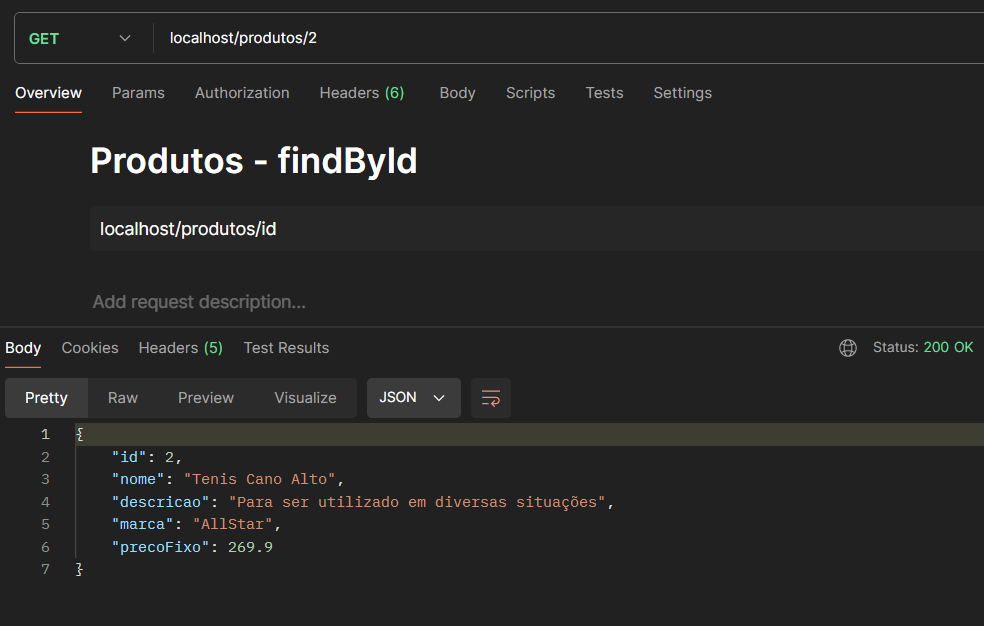
**Endpoint:** localhost/produtos?marca=AllStar



**Endpoint:** localhost/produtos?preco=250.99

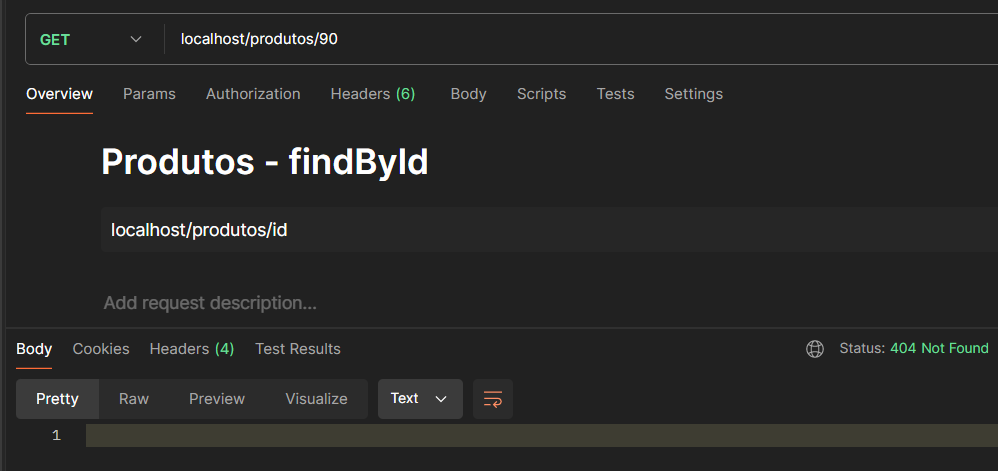


**Endpoint:** localhost/produtos/2



**Endpoint:** localhost/produtos/90

Ao colocar um id que não existe ele retorna 404.

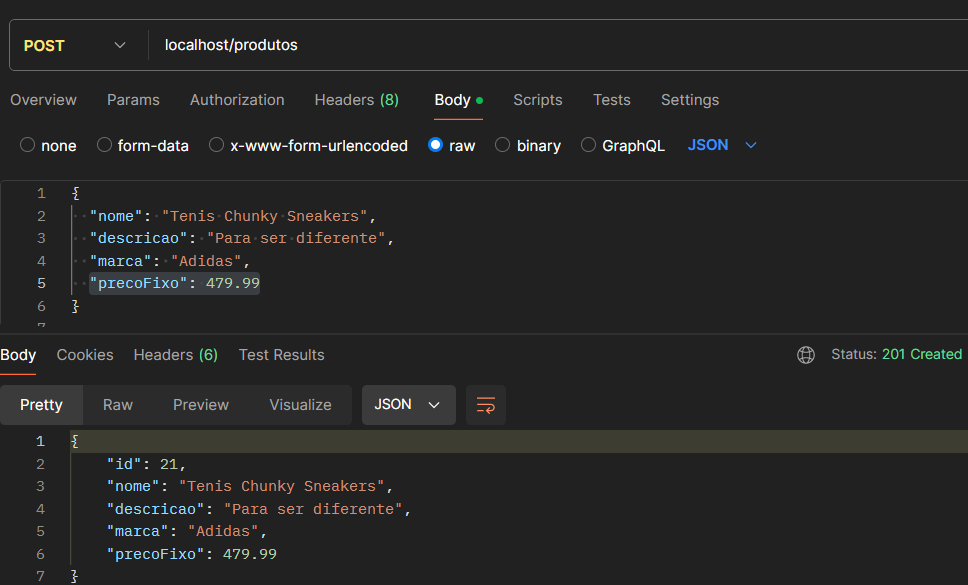


**Produtos – POST**

**Endpoint:** localhost/produtos

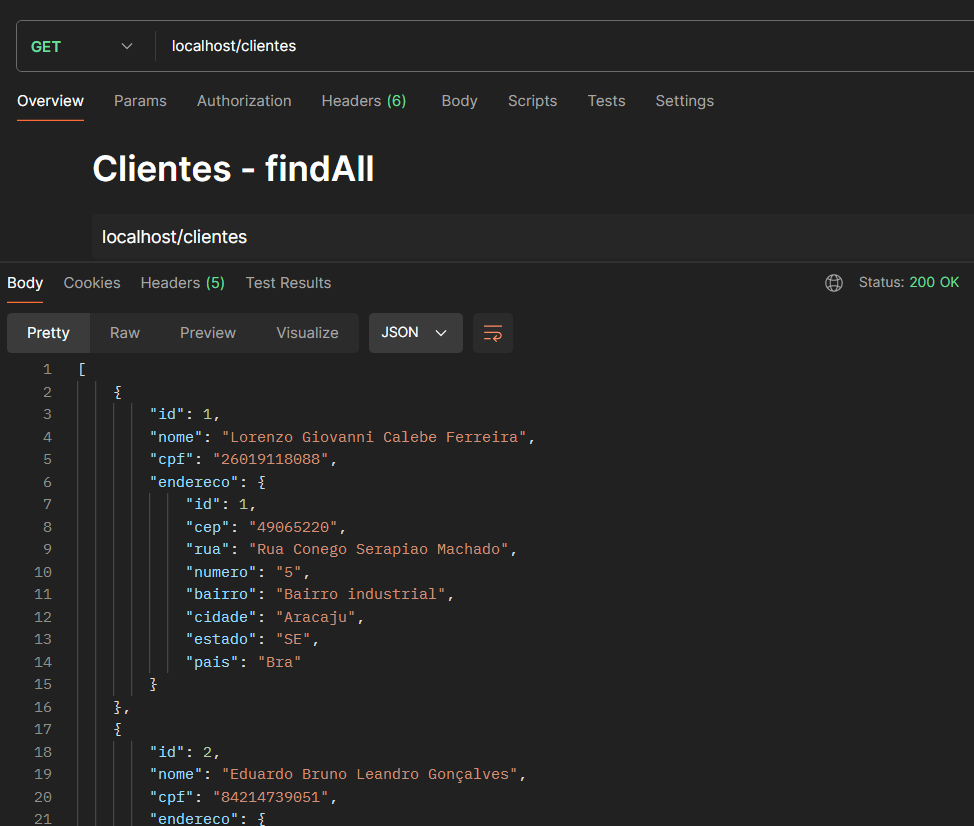
Para salvar os produtos nenhuma informação pode estar em falta ou retornará um erro 400.

|  |  |
| --- | --- |
|  |  |
|  |  |

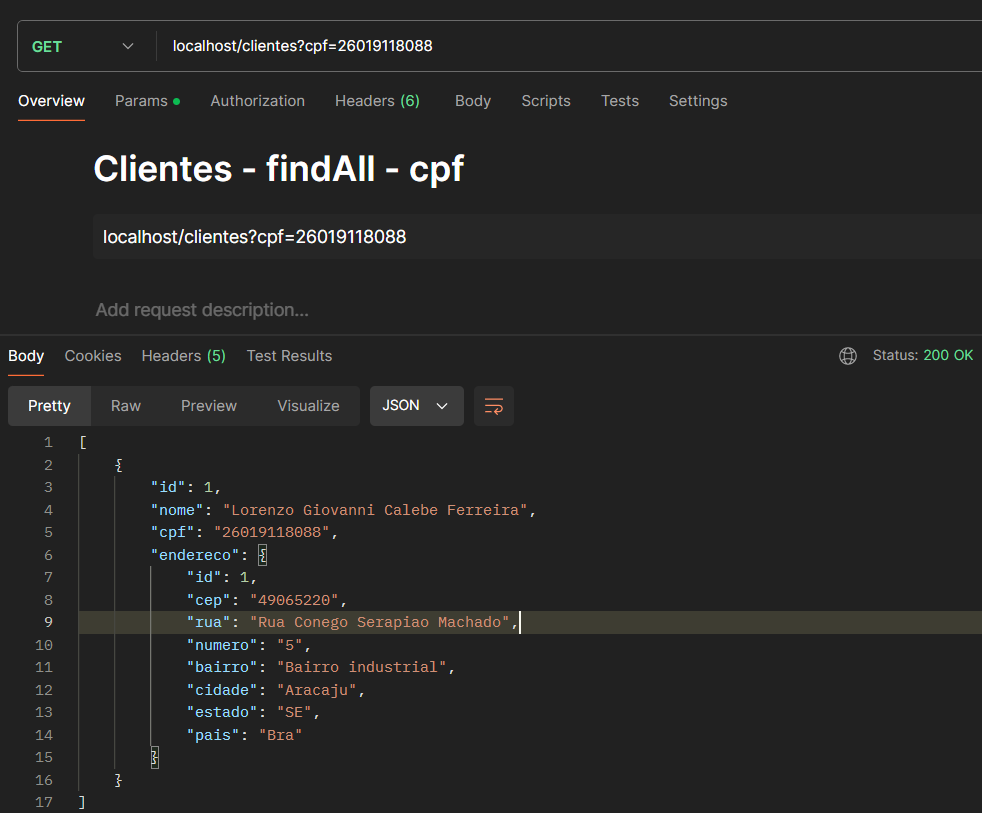


**Clientes – GET**

**Endpoint:** localhost/clientes



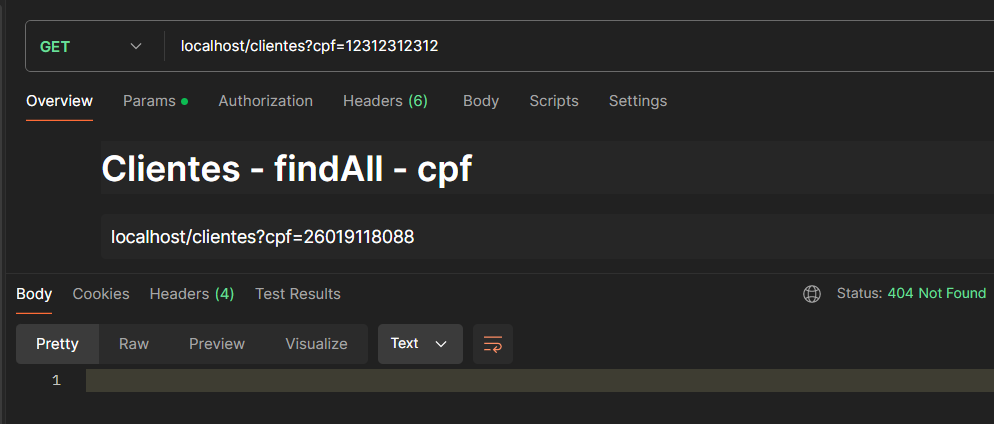
**Endpoint:** localhost/clientes?cpf= 26019118088



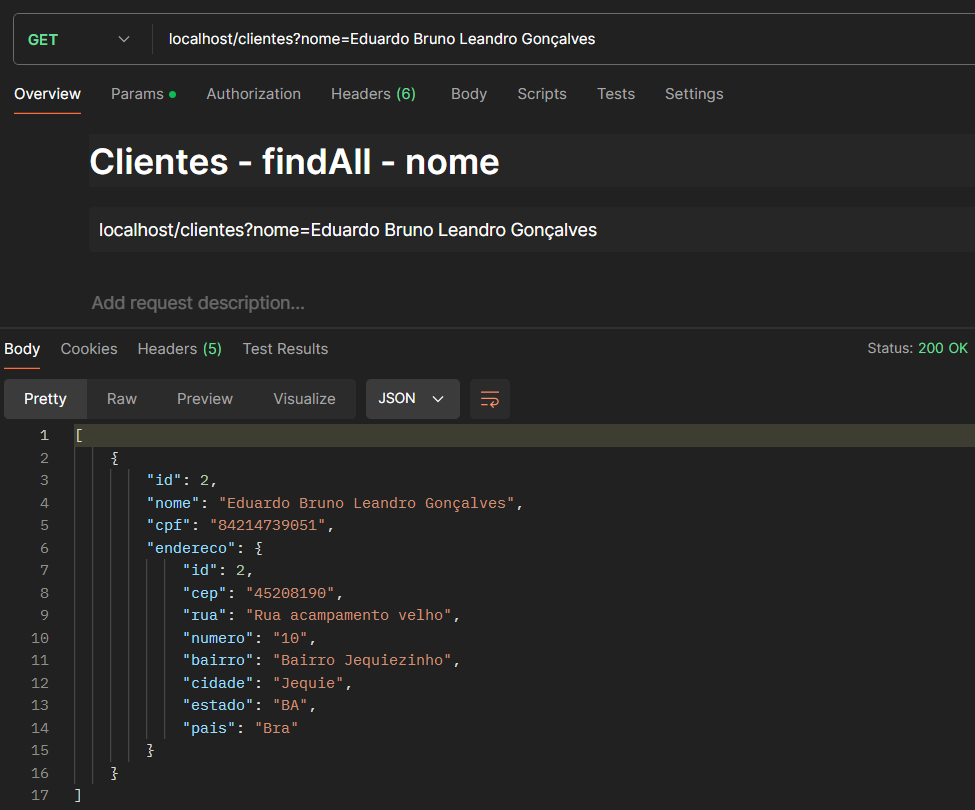
**Endpoint:** localhost/clientes?cpf=12312312312

Buscar um cliente por um CPF que não existe, a resposta será 404.

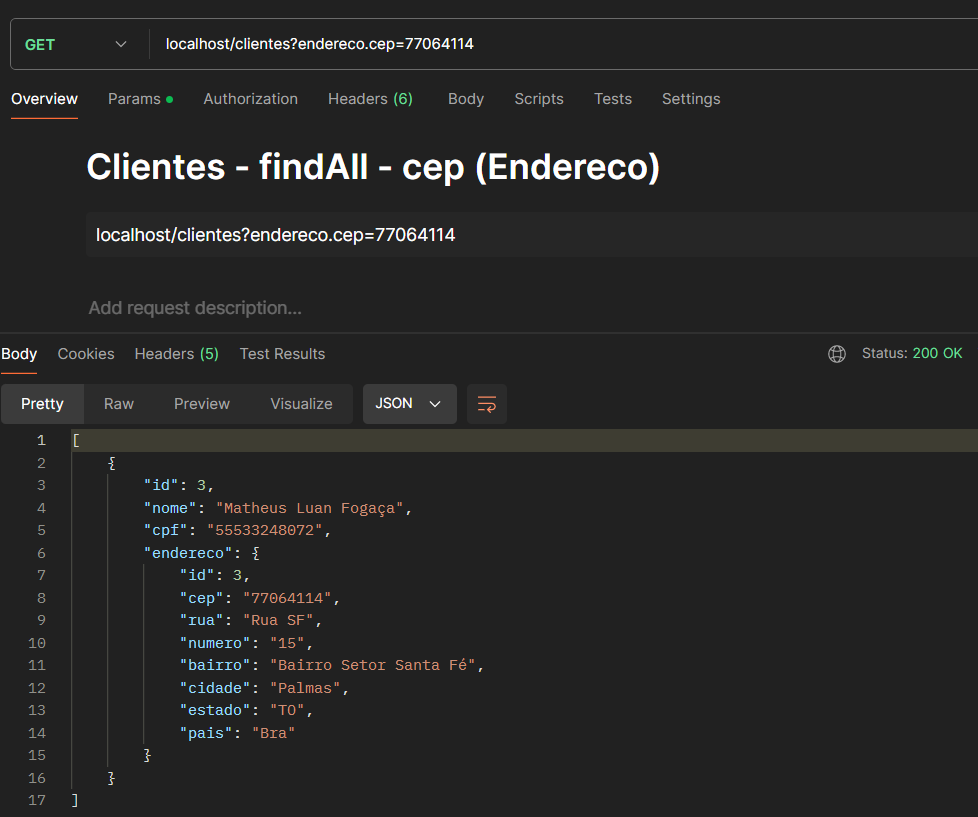
O mesmo ocorre para todos os endpoints de GET por findAll(Example)



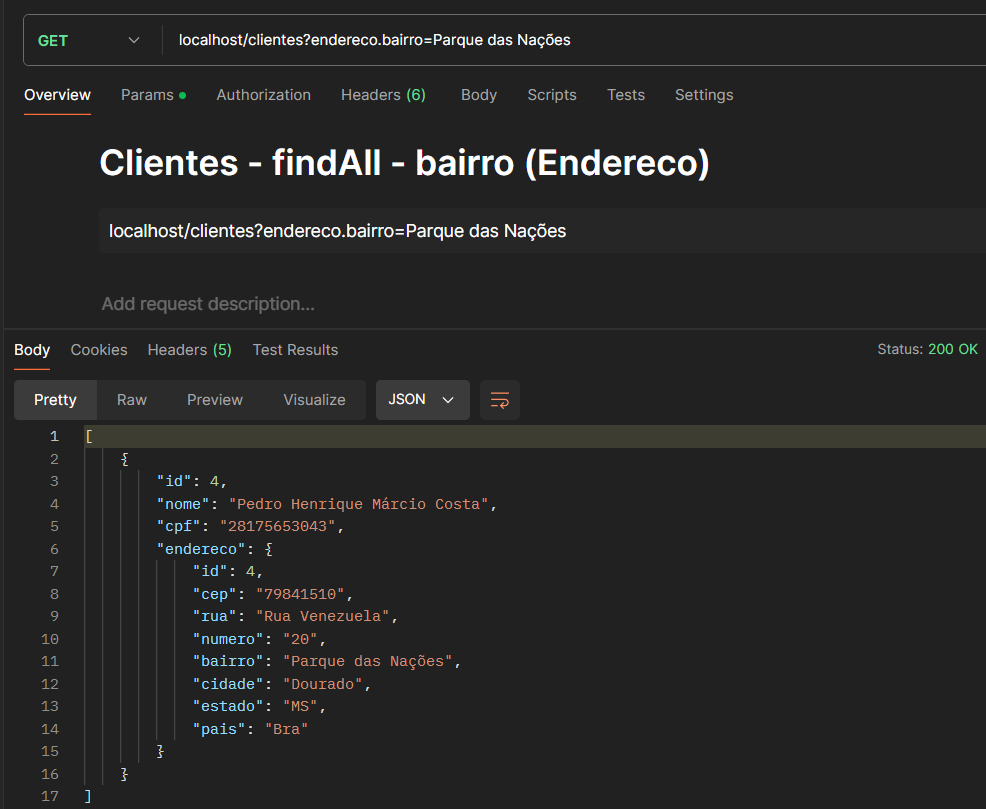
**Endpoint:** localhost/clientes?nome=Eduardo Bruno Leandro Gonçalves



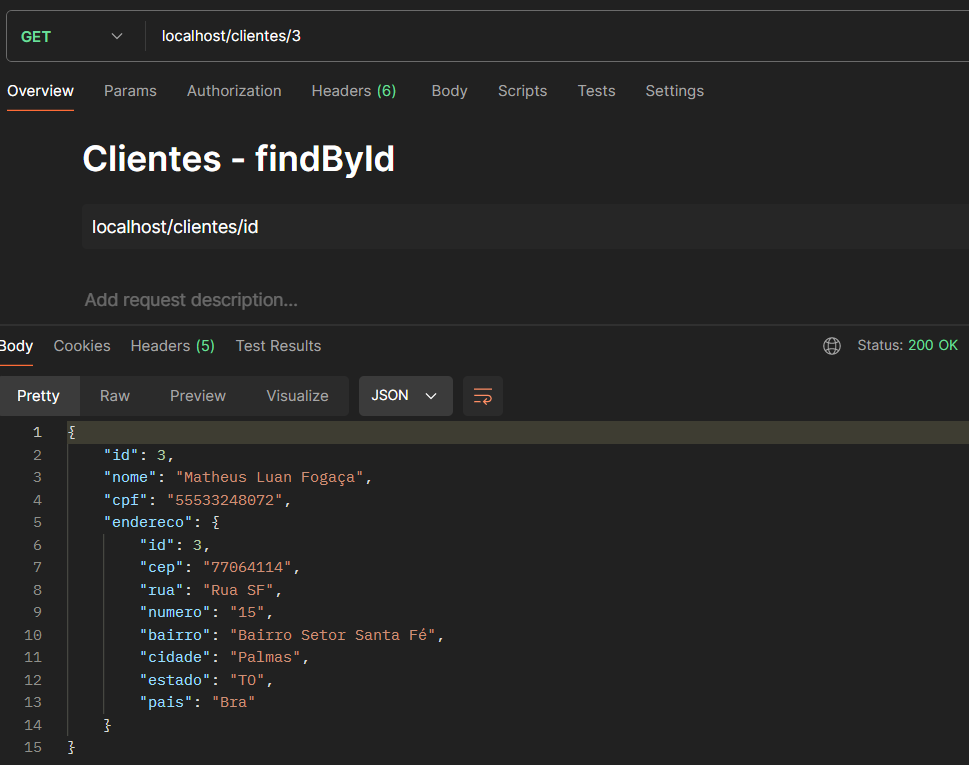
**Endpoint:** localhost/clientes?endereco.cep=77064114 – busca pelo CEP associado ao cliente



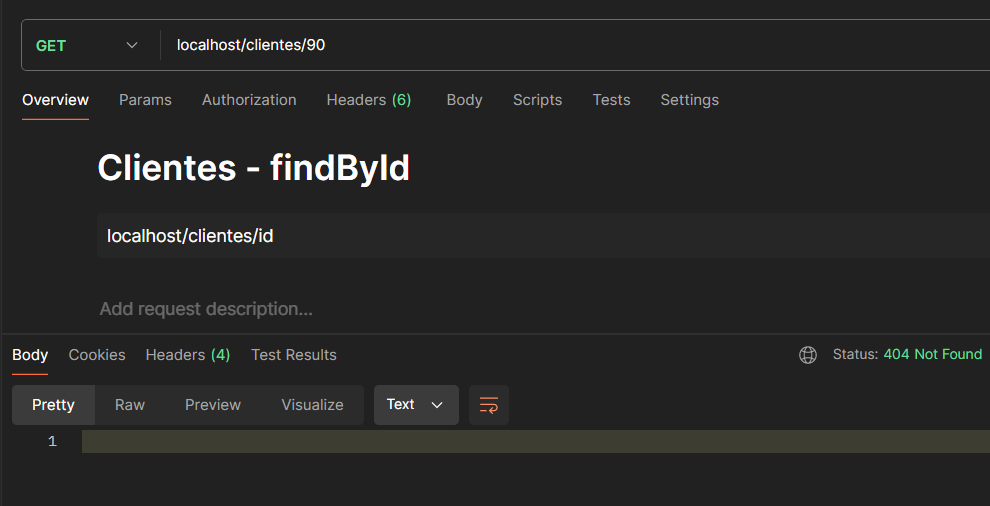
**Endpoint:** localhost/clientes?endereco.bairro=Parque das Nações



**Endpoint:** localhost/clientes/3

****

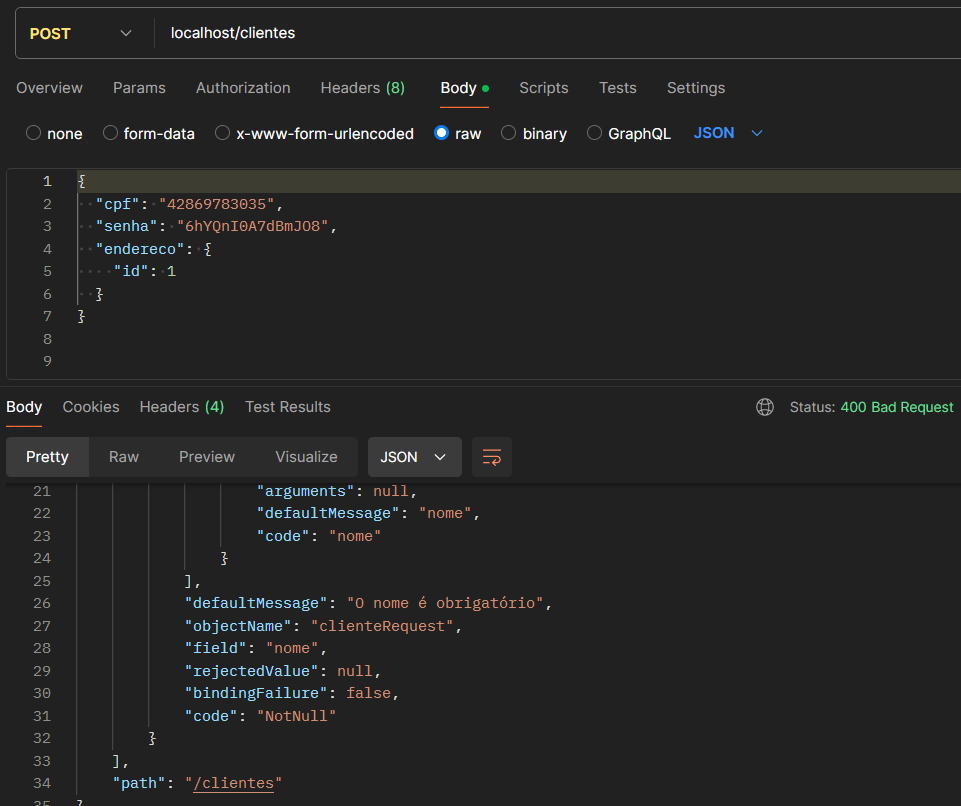
Ao colocar um id que não existe ele retorna 404.



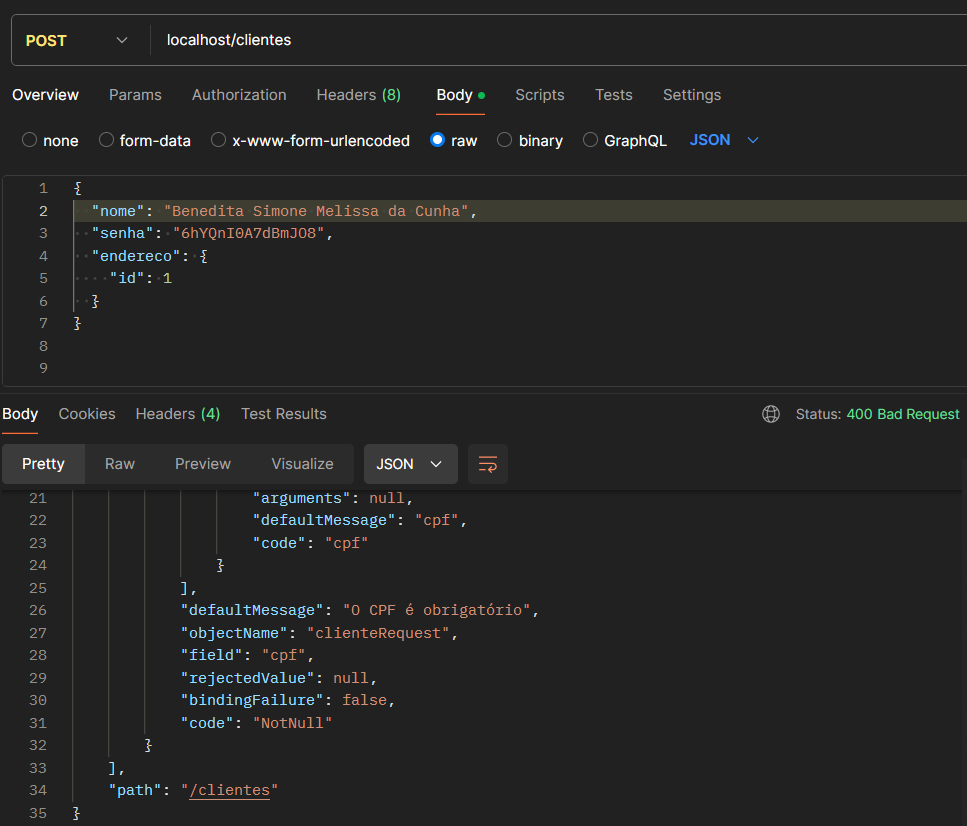
**Clientes – POST**

Para salvar um cliente novo temos uma série de verificações para a validação dos dados.

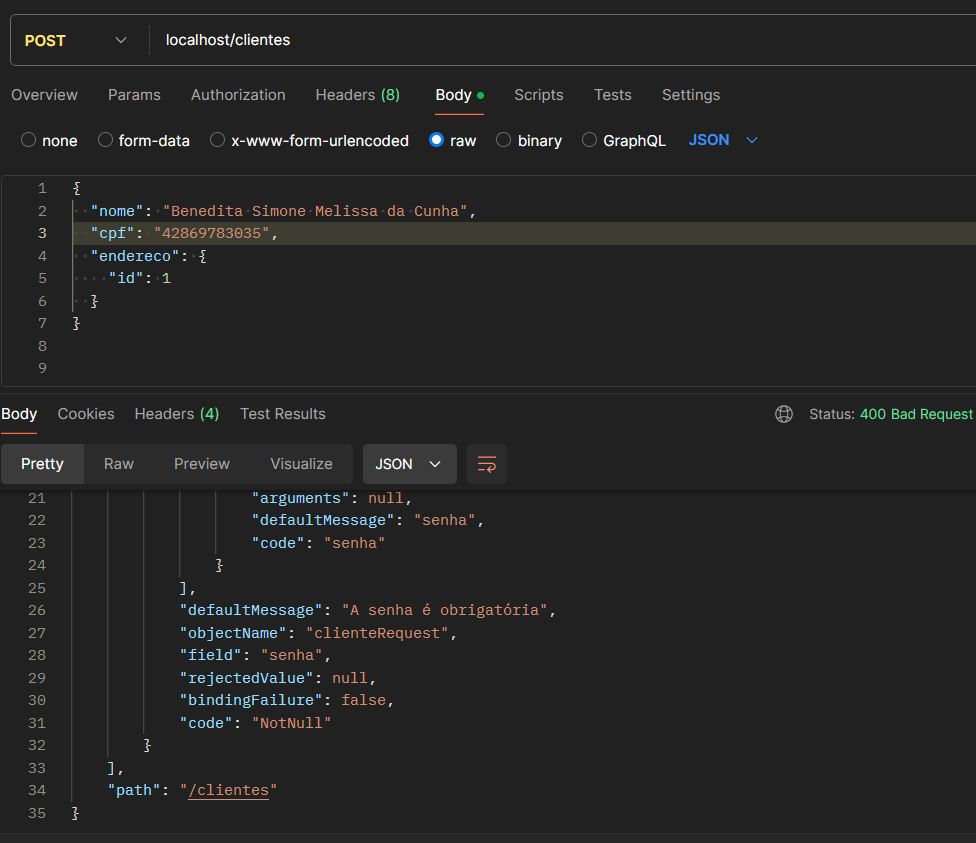
1 - O nome deve ser incluído



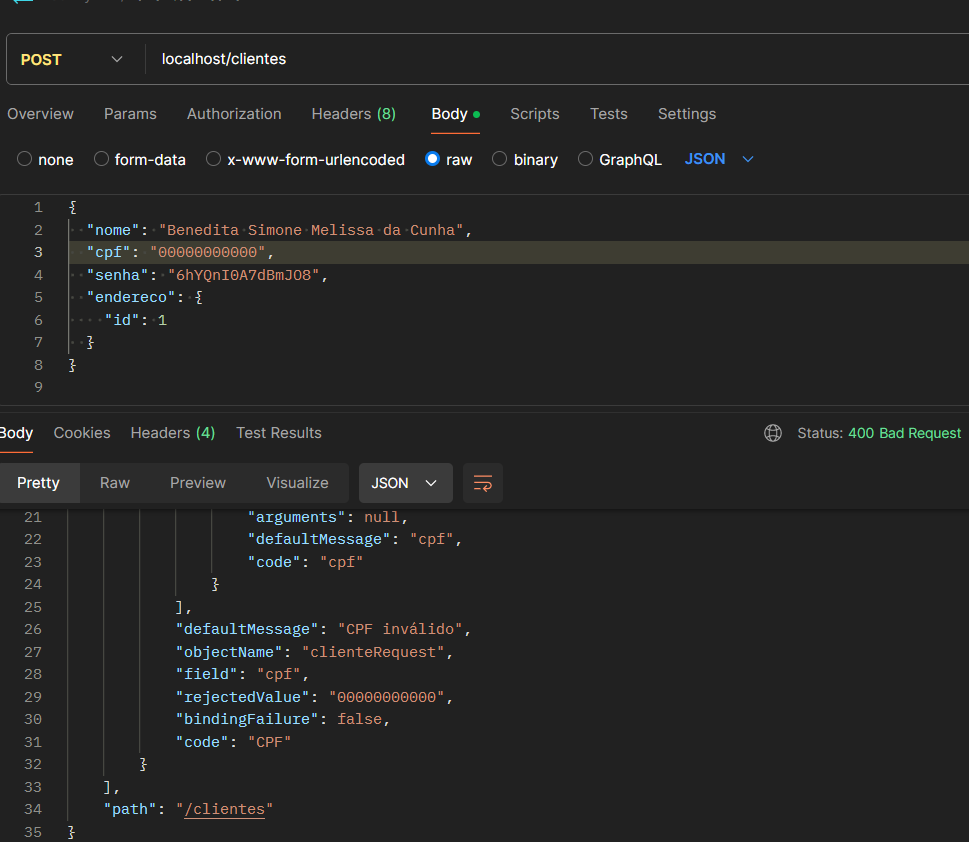
2 – O CPF deve ser incluído



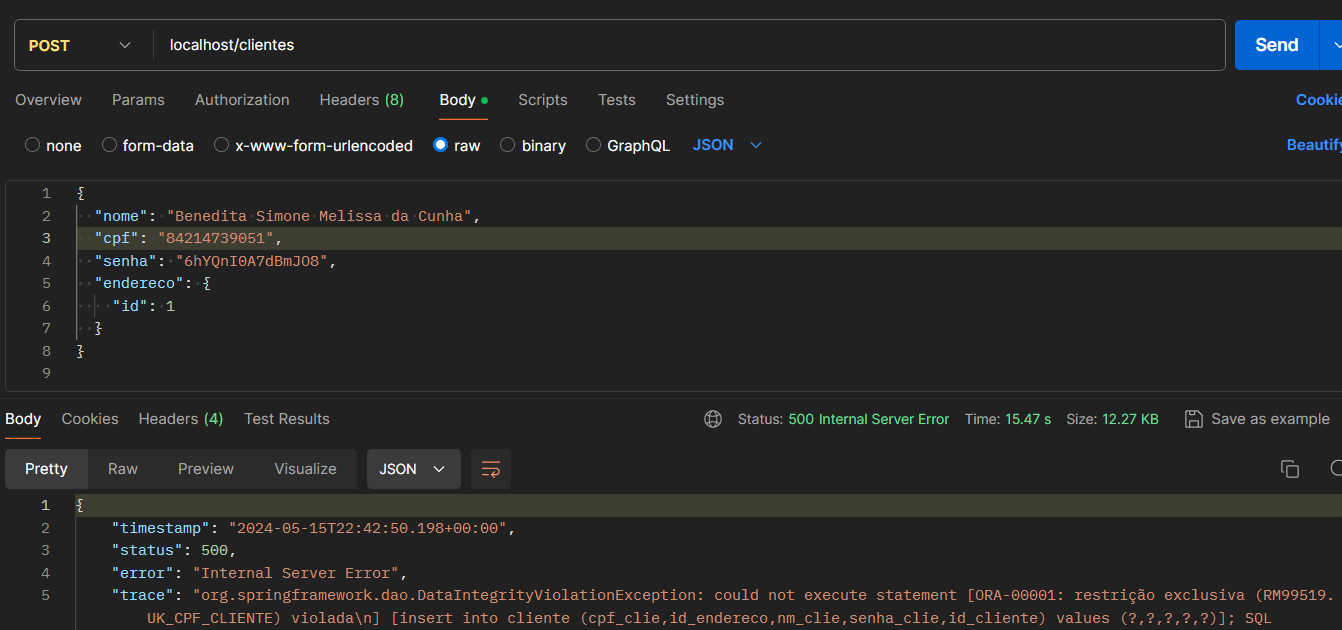
3 – A senha deve ser incluída

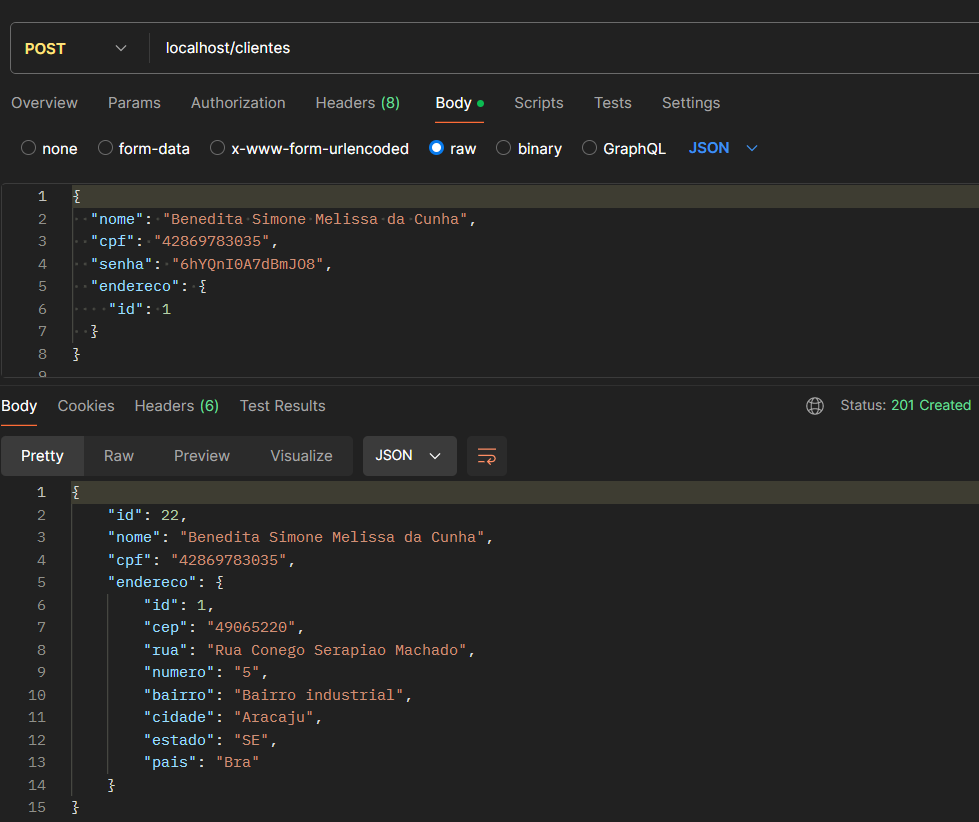


4 – O CPF deve ser válido



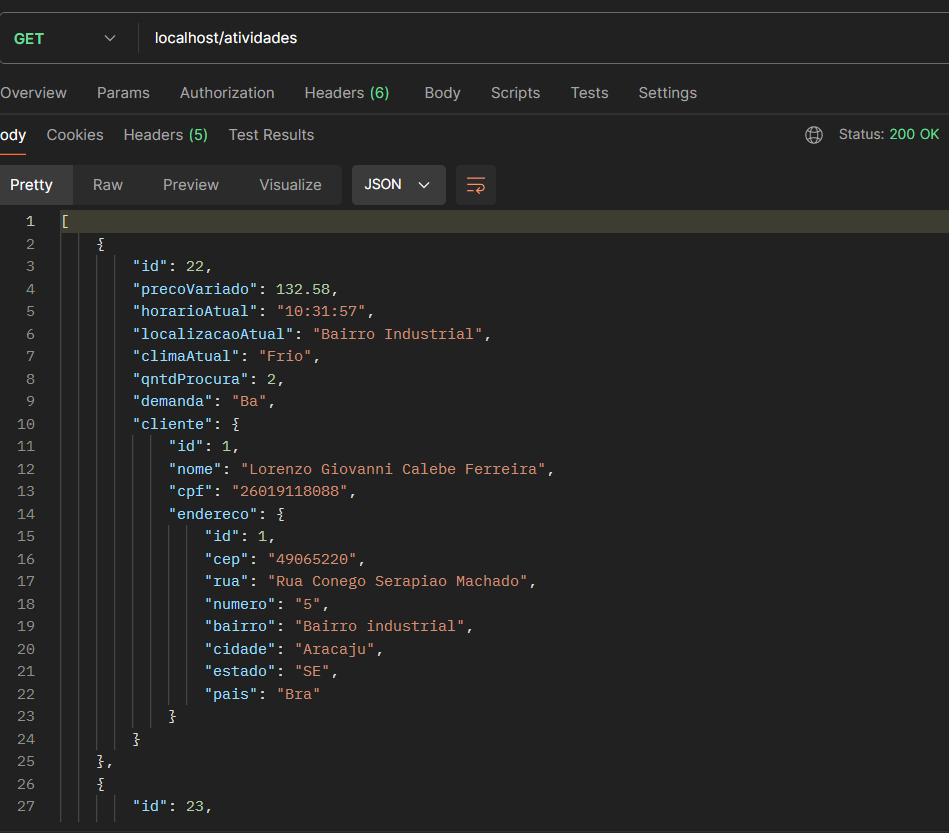
5 – O CPF não pode existir em outra conta



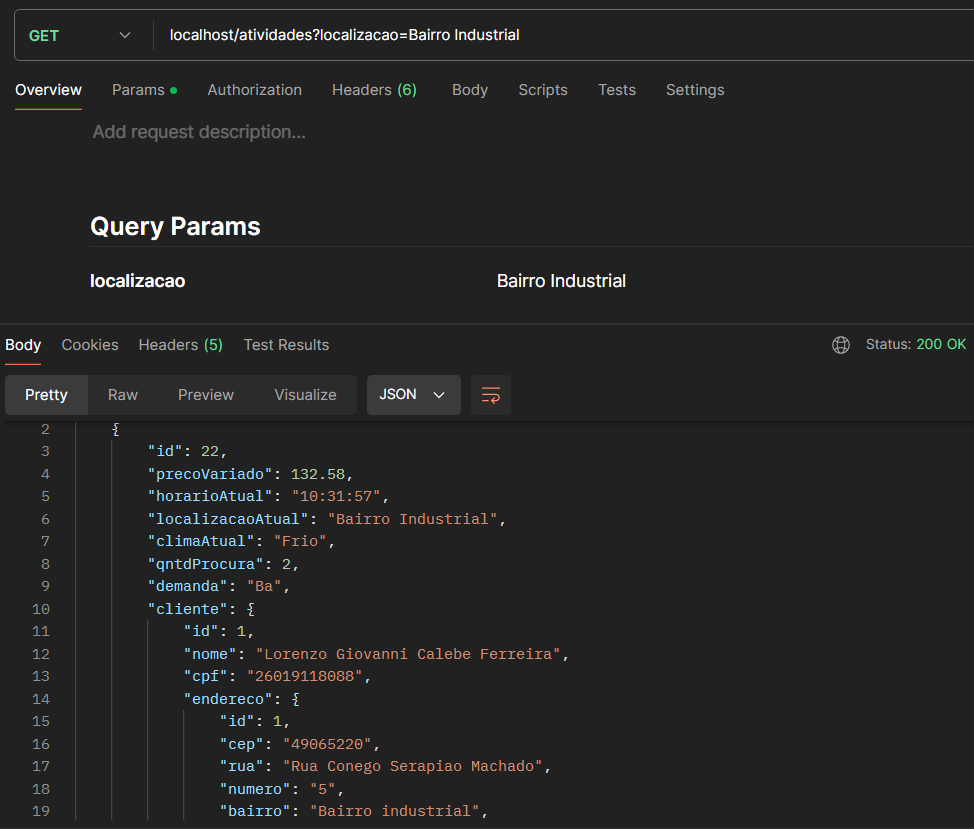


**Atividades – GET**

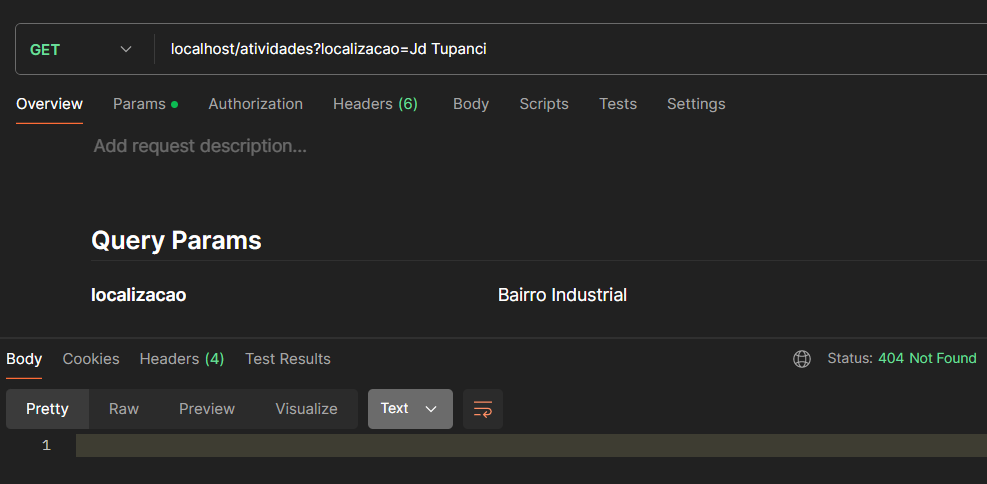
**Endpoint:** localhost/atividades



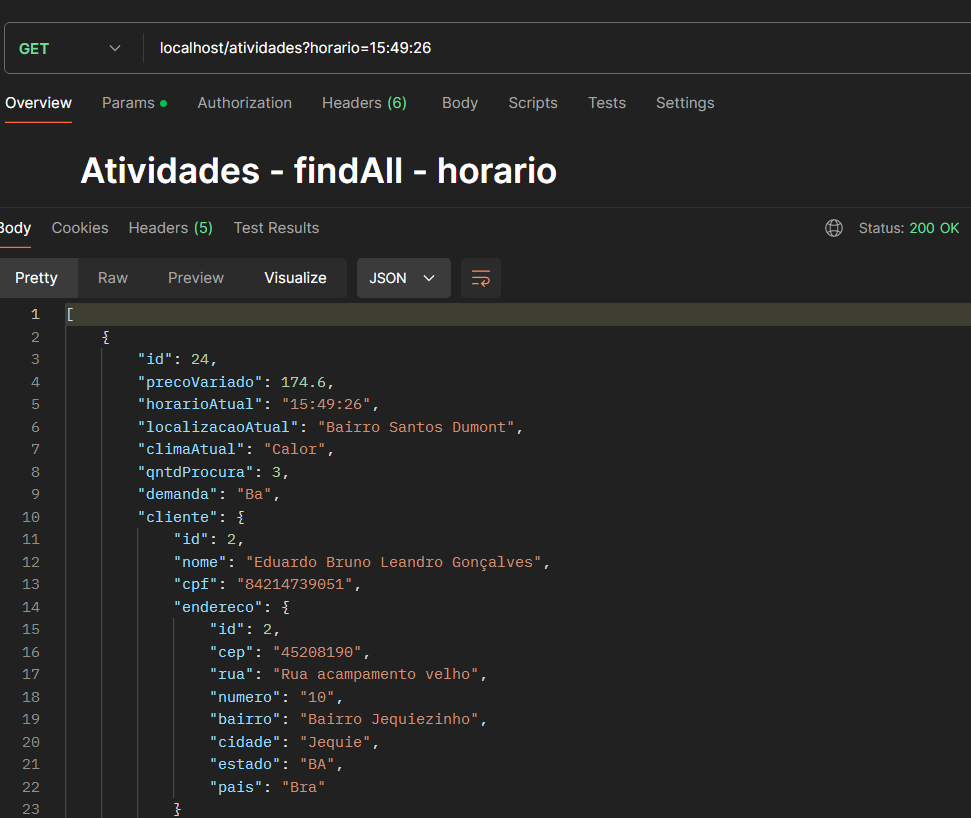
**Endpoint:** localhost/atividades?localizacao=Bairro Industrial



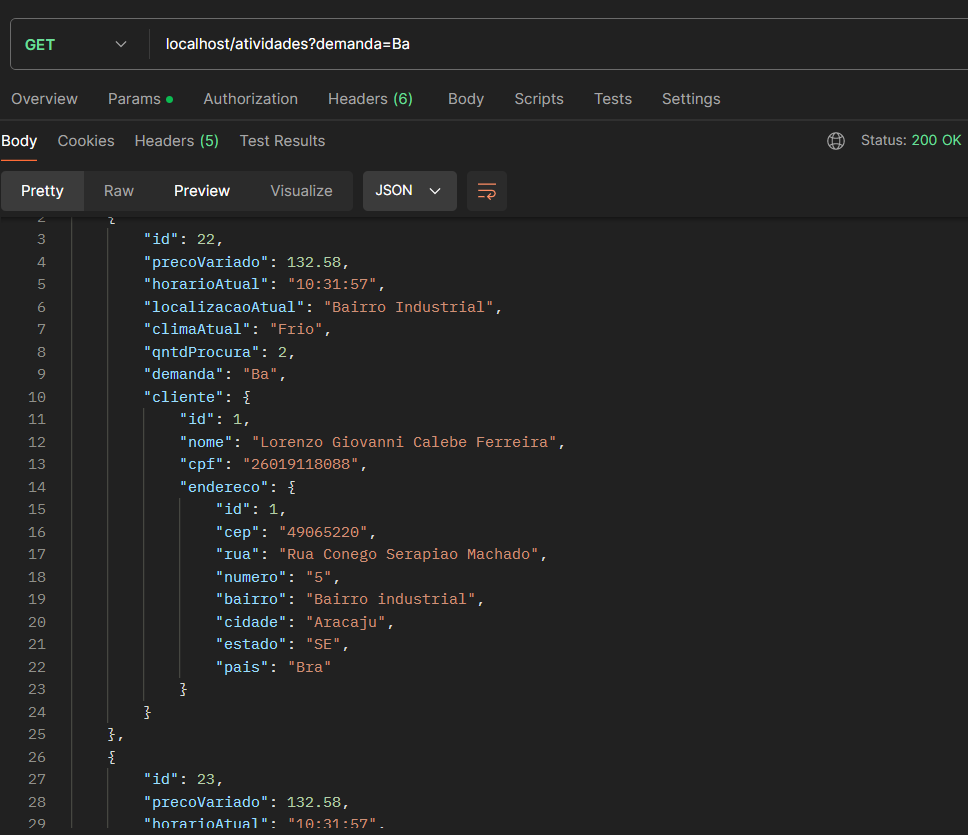
Caso o bairro não exista ele retorna um 404. O mesmo ocorre para todos os endpoints de GET por findAll(Example)



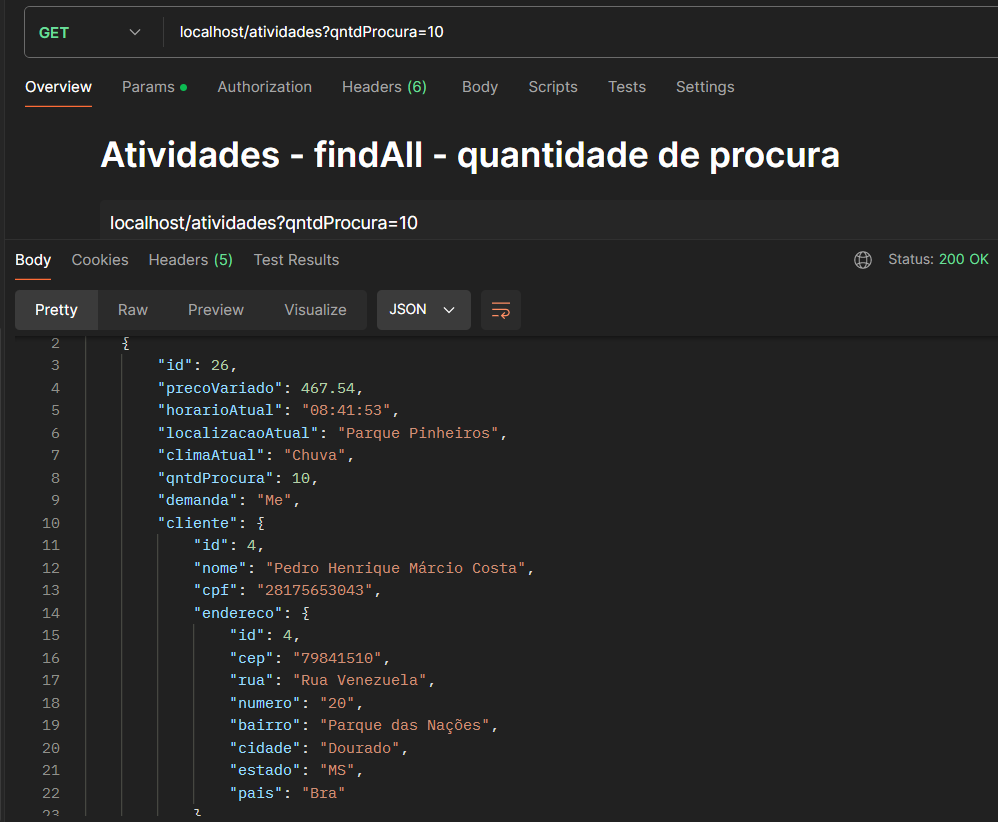
**Endpoint:** localhost/atividades?horario=15:49:26



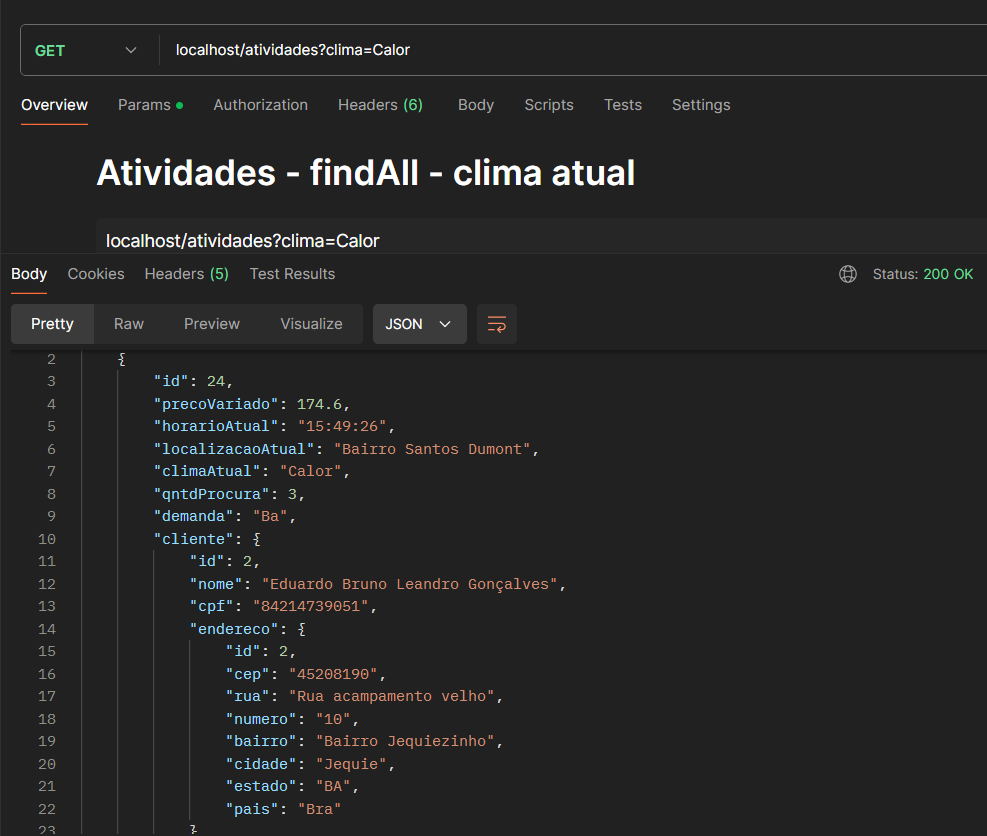
**Endpoint:** localhost/atividades?demanda=Ba



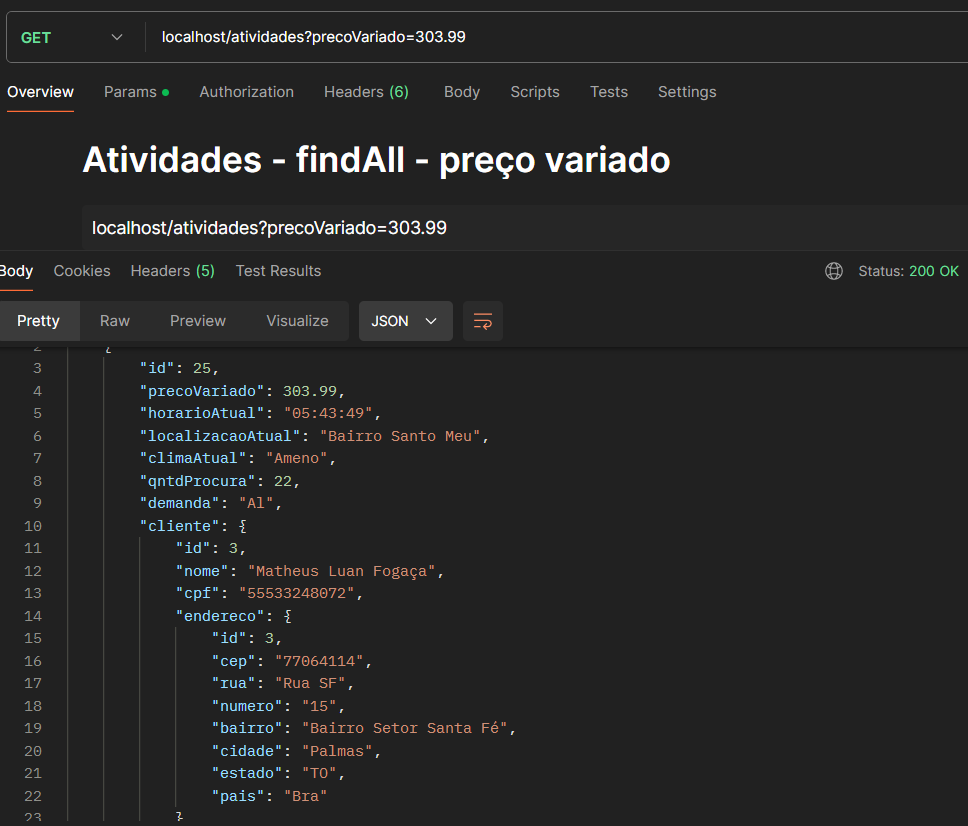
**Endpoint:** localhost/atividades?qntdProcura=10



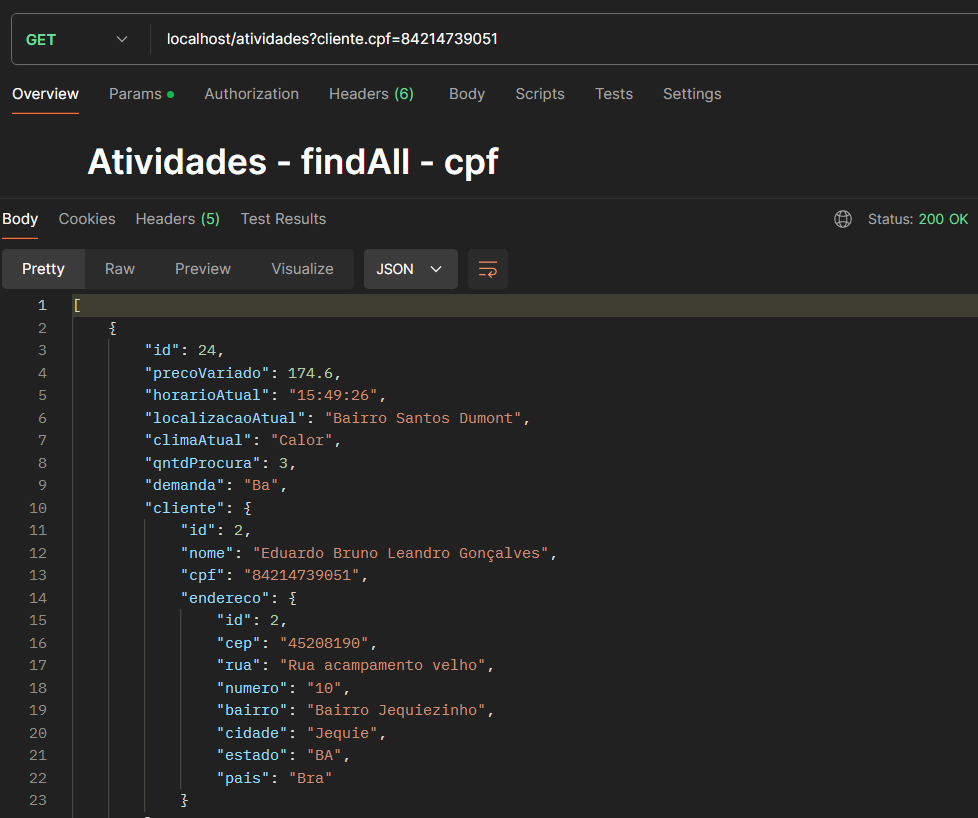
**Endpoint:** localhost/atividades?clima=Calor



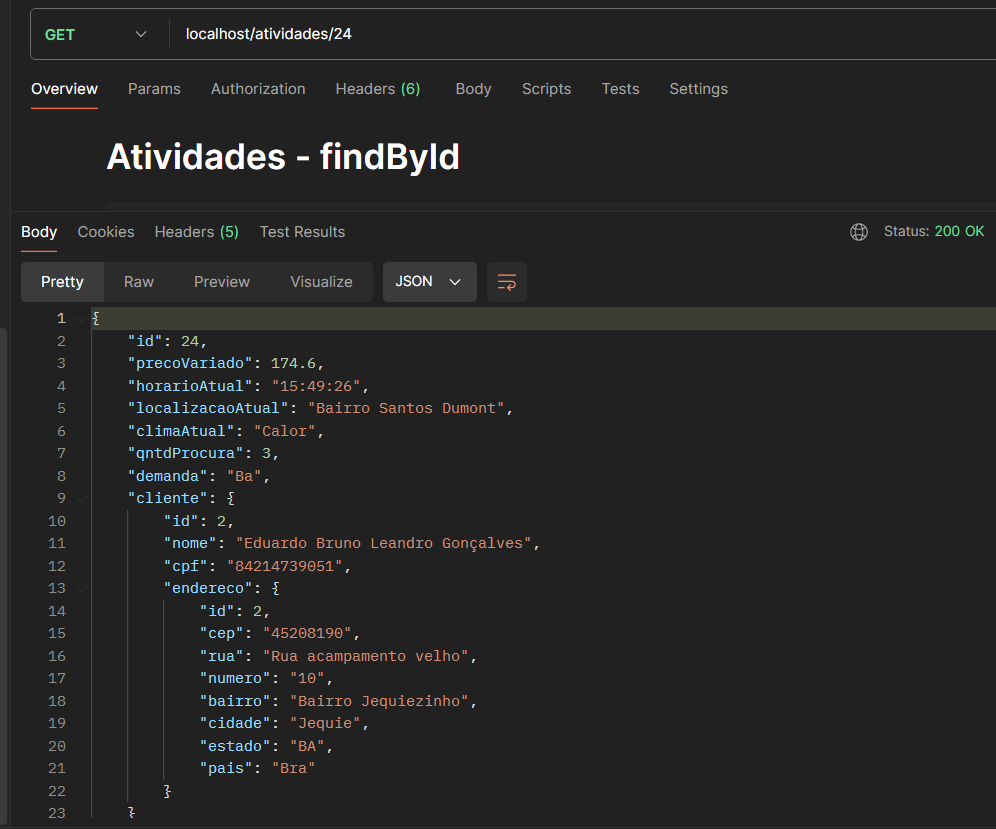
**Endpoint:** localhost/atividades?precoVariado=303.99



**Endpoint:** localhost/atividades?cliente.cpf=84214739051 – busca pelo CPF do cliente associado

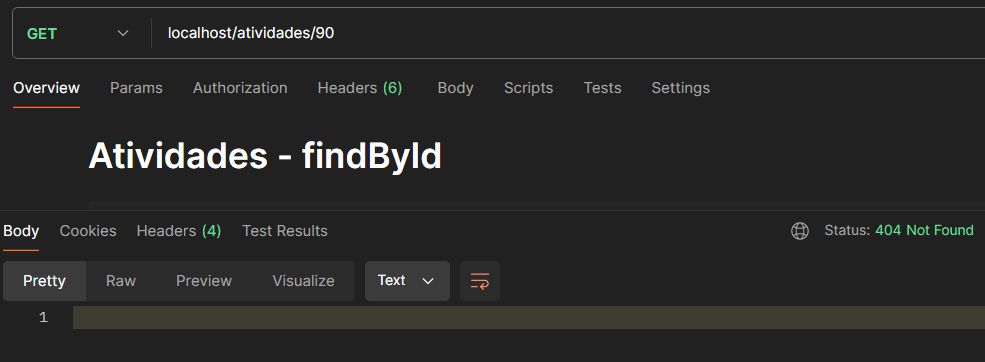


**Endpoint:** localhost/atividades/24



**Endpoint:** localhost/atividades/90

Ao colocar um id que não existe ele retorna 404.

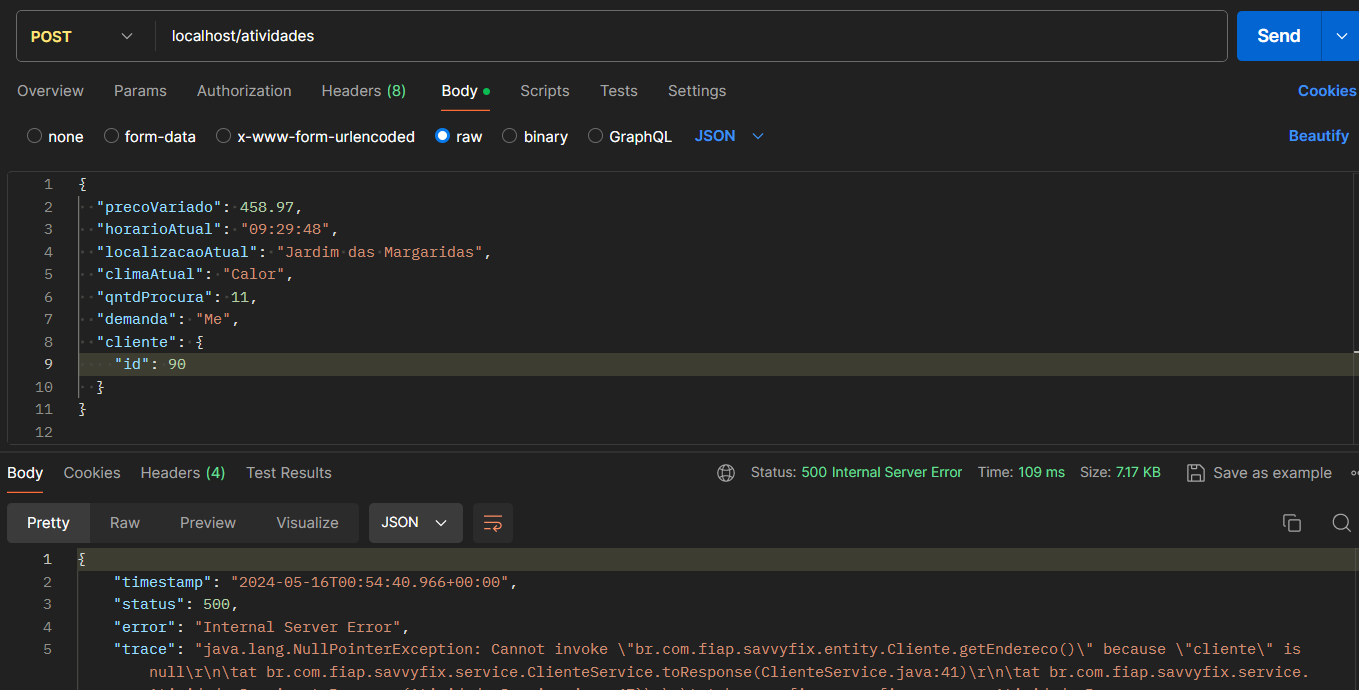


**Atividades – POST**

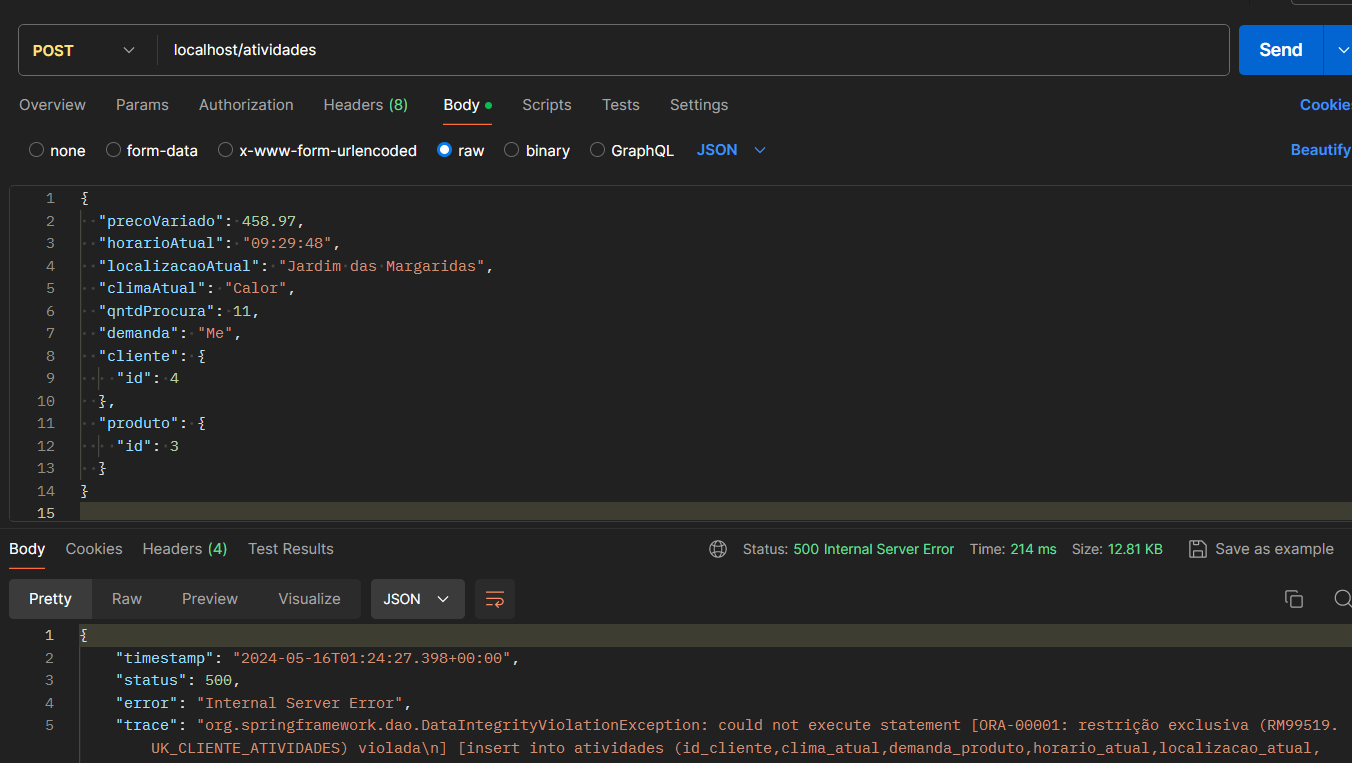
**Endpoint:** localhost/atividades

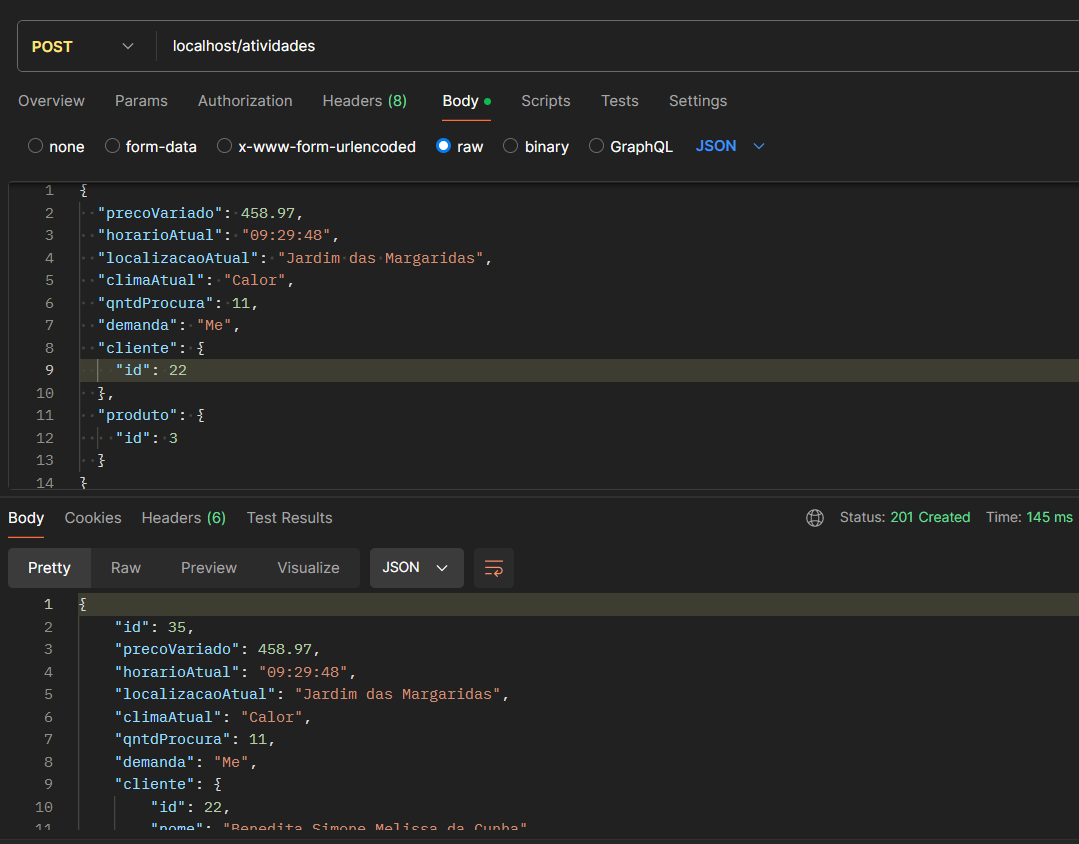
Para salvar os dados é necessário:

1 – O Id do cliente ser válido

****

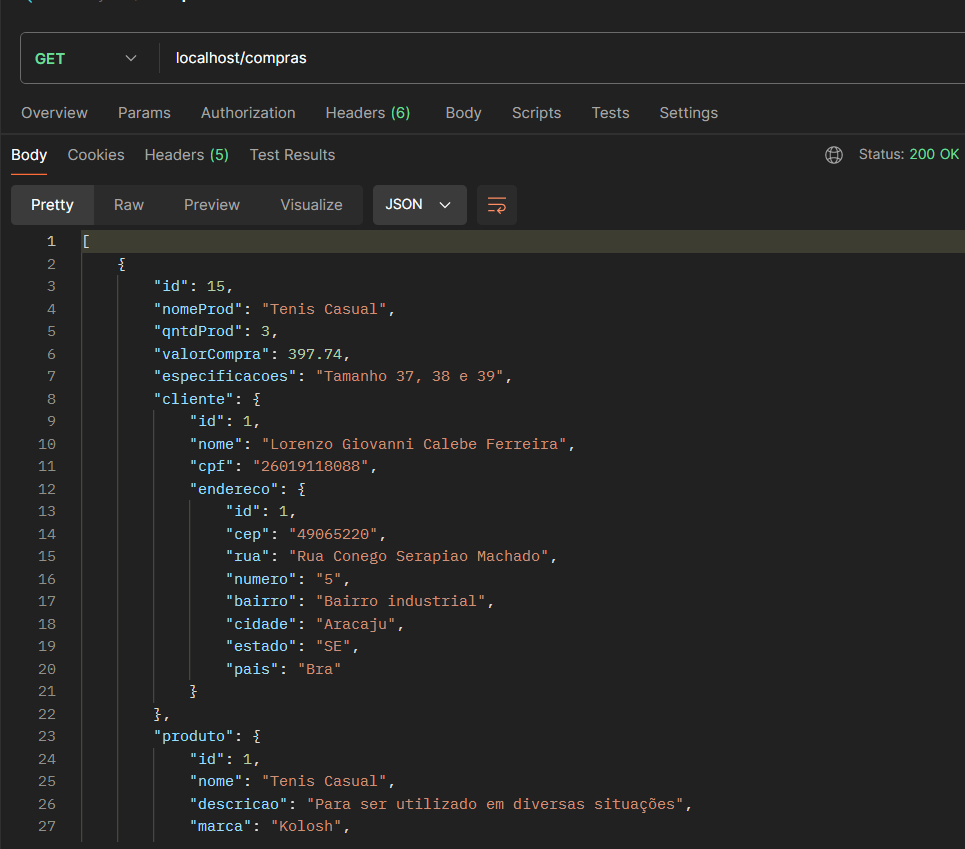
2 – O Id do cliente não pode se repetir

****

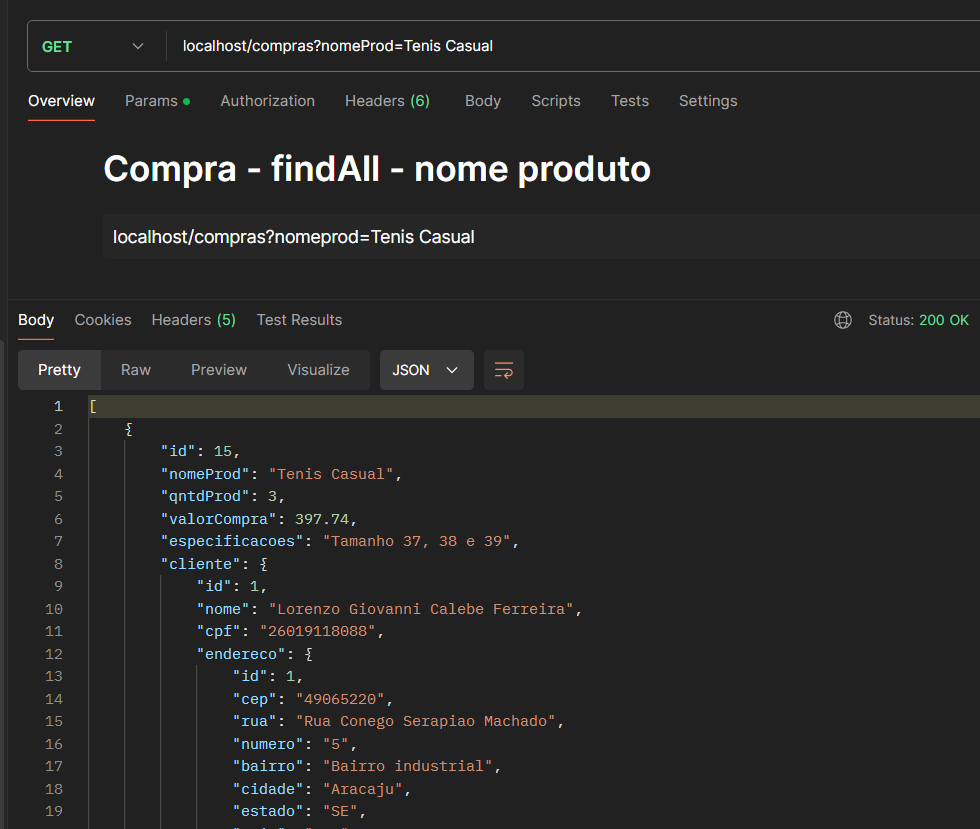
****

**Compras – GET**

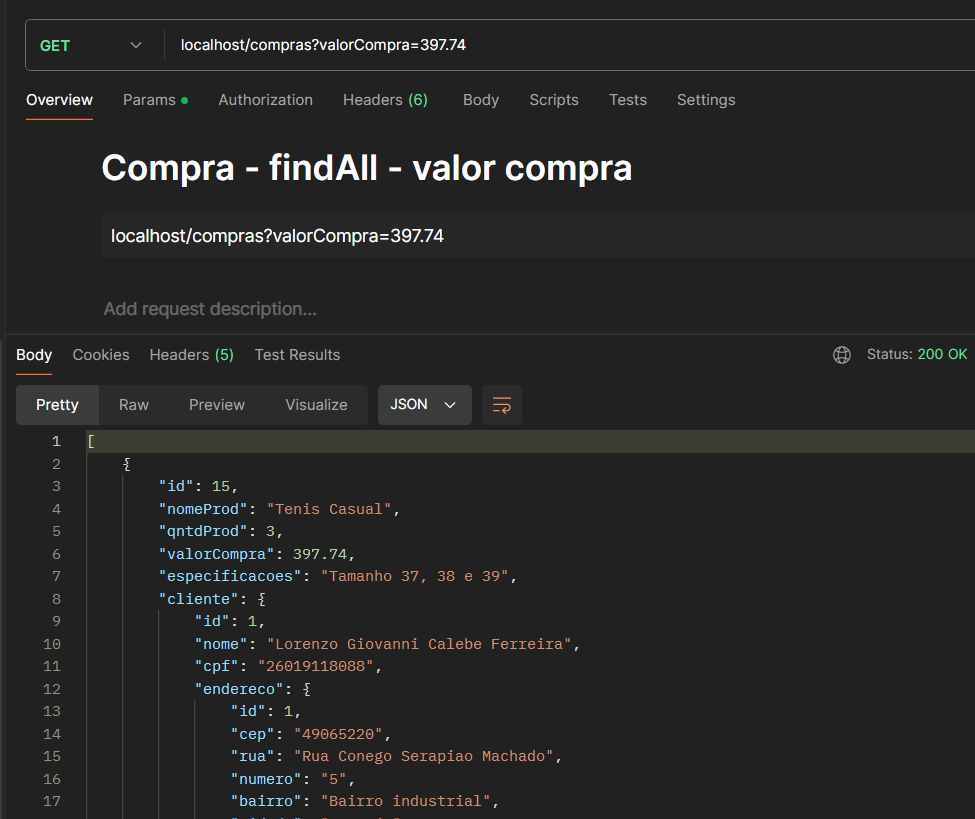
**Endpoint:** localhost/compras



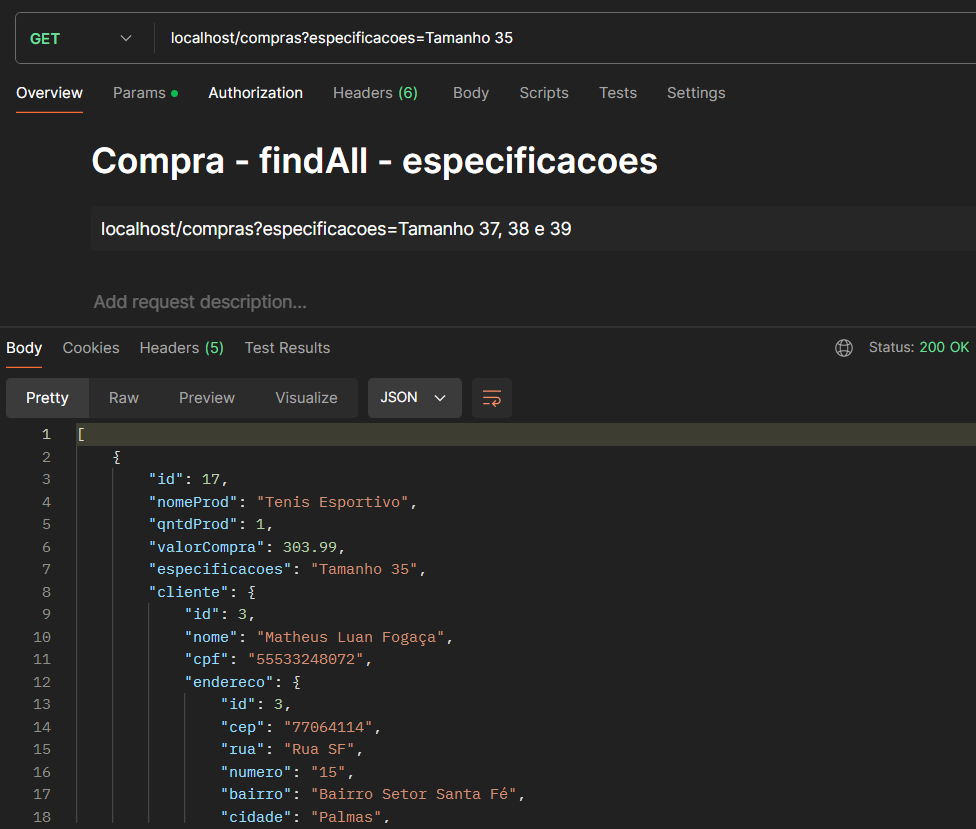
**Endpoint:** localhost/compras?nomeProd=Tenis Casual



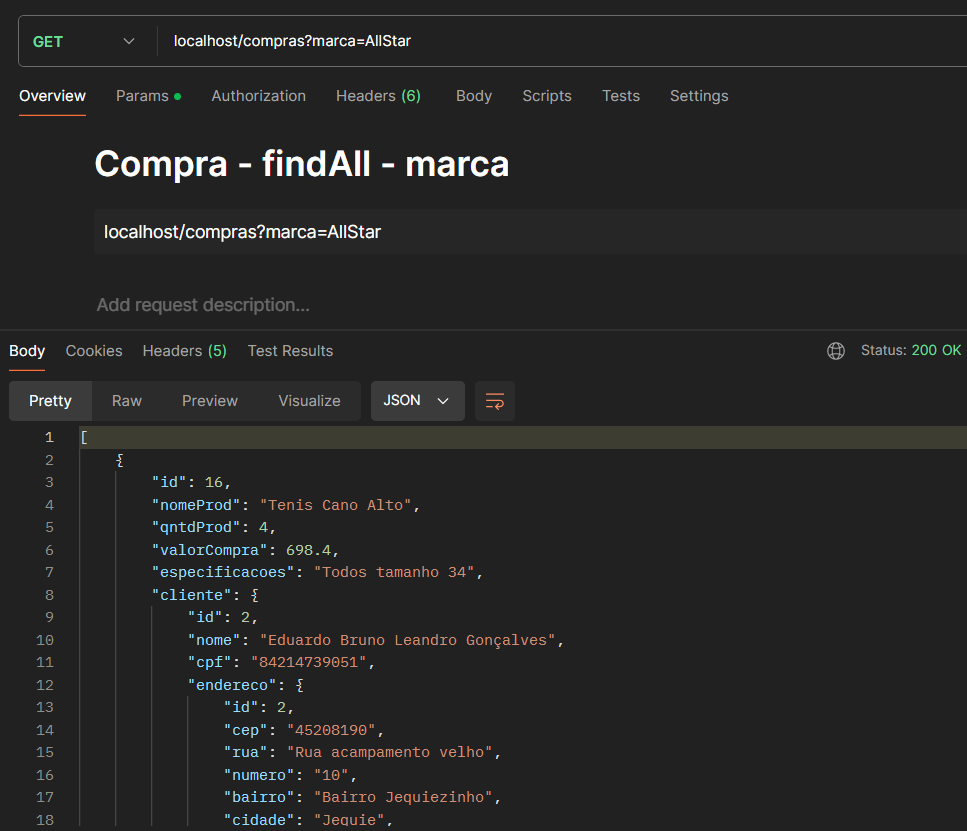
**Endpoint:** localhost/compras?valorCompra=397.74



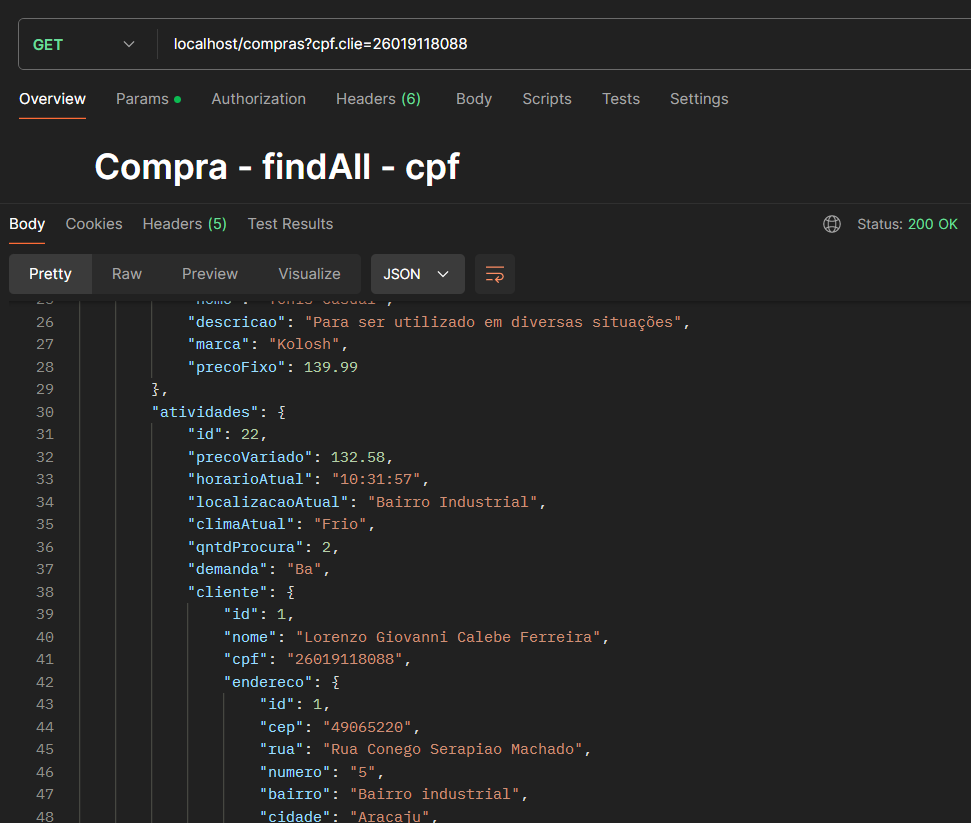
**Endpoint:** localhost/compras?especificacoes=Tamanho 35



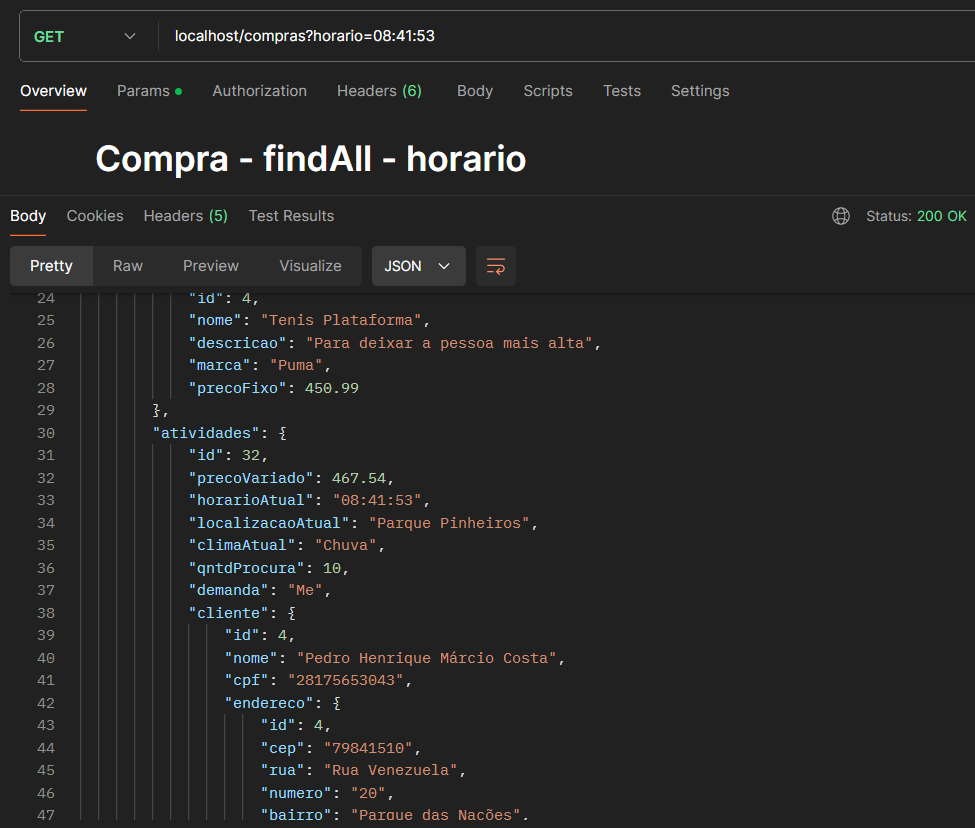
**Endpoint:** localhost/compras?marca=AllStar



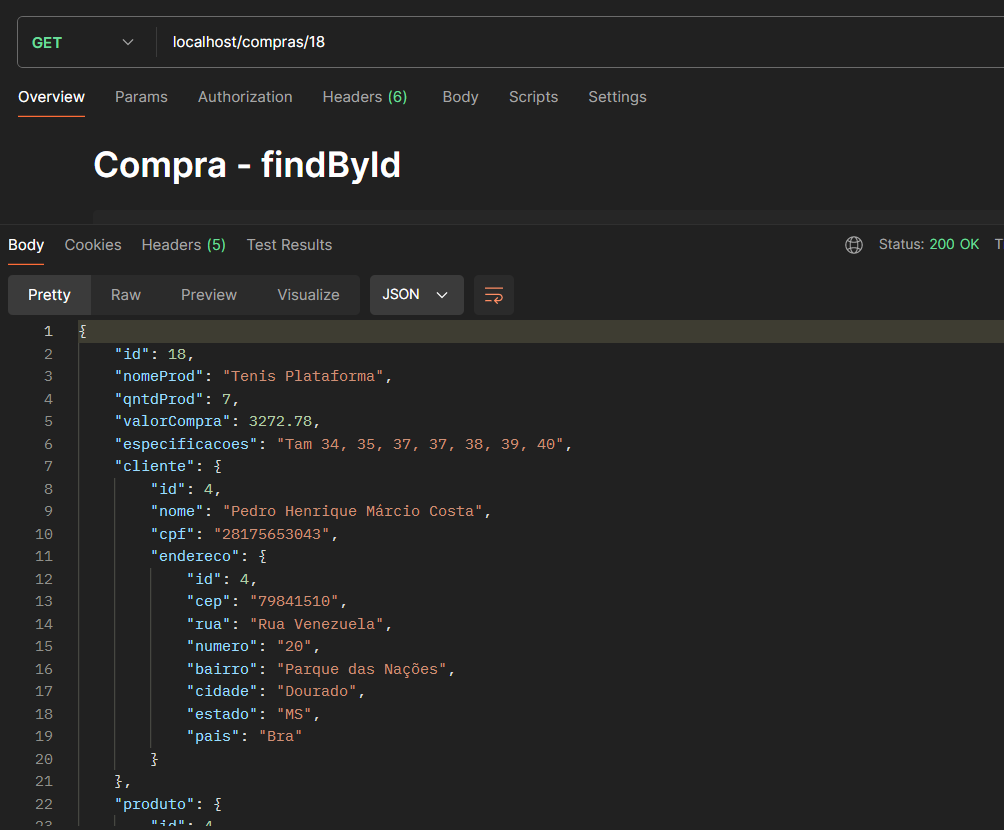
**Endpoint:** localhost/compras?cpf.clie=26019118088 – busca pelo CPF do cliente associado a atividade em compra



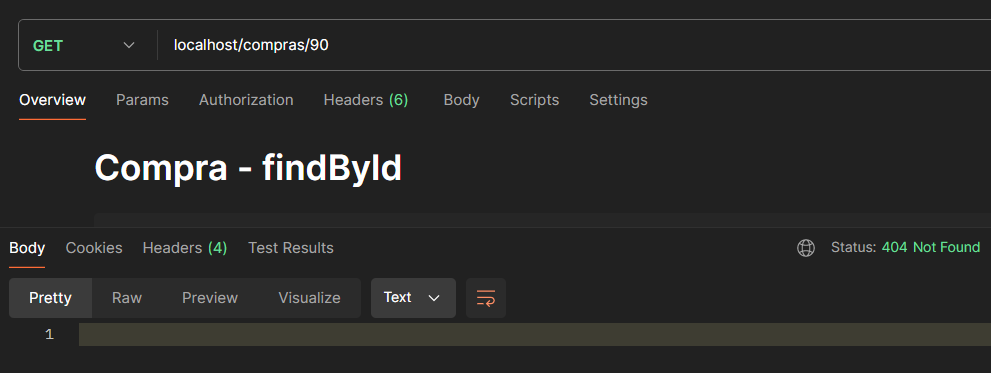
**Endpoint:** localhost/compras?horario=08:41:53



**Endpoint:** localhost/compras/18



Ao colocar um id que não existe ele retorna 404.

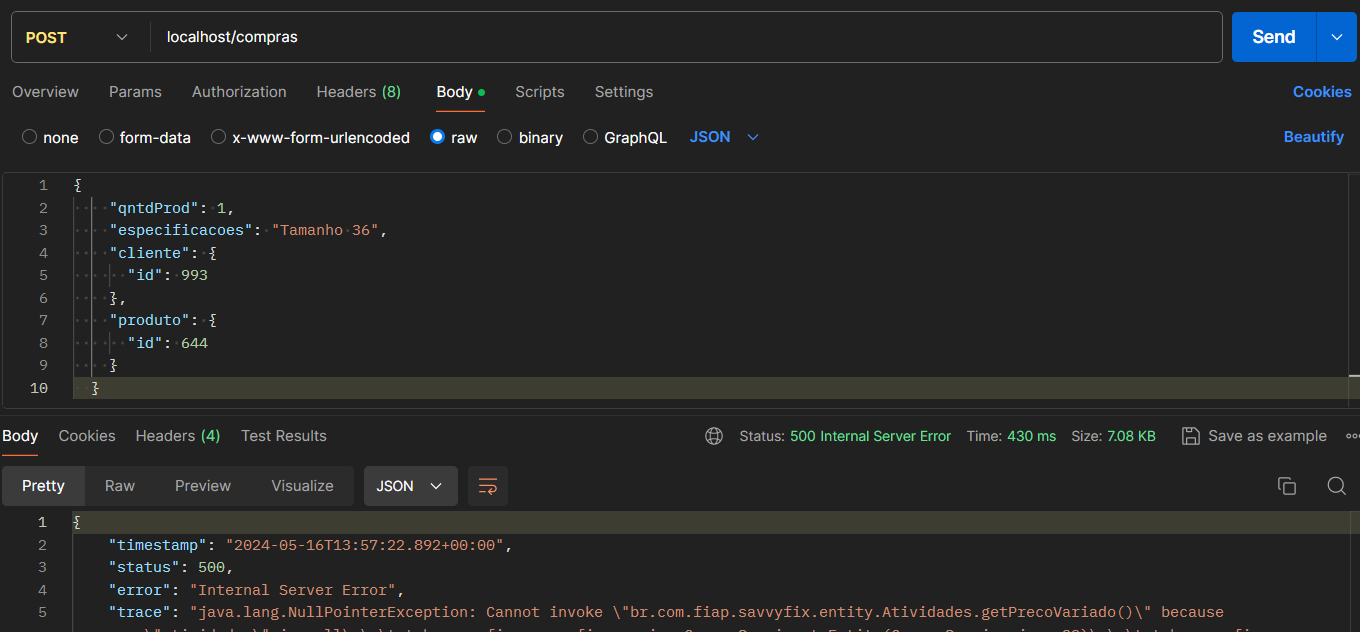


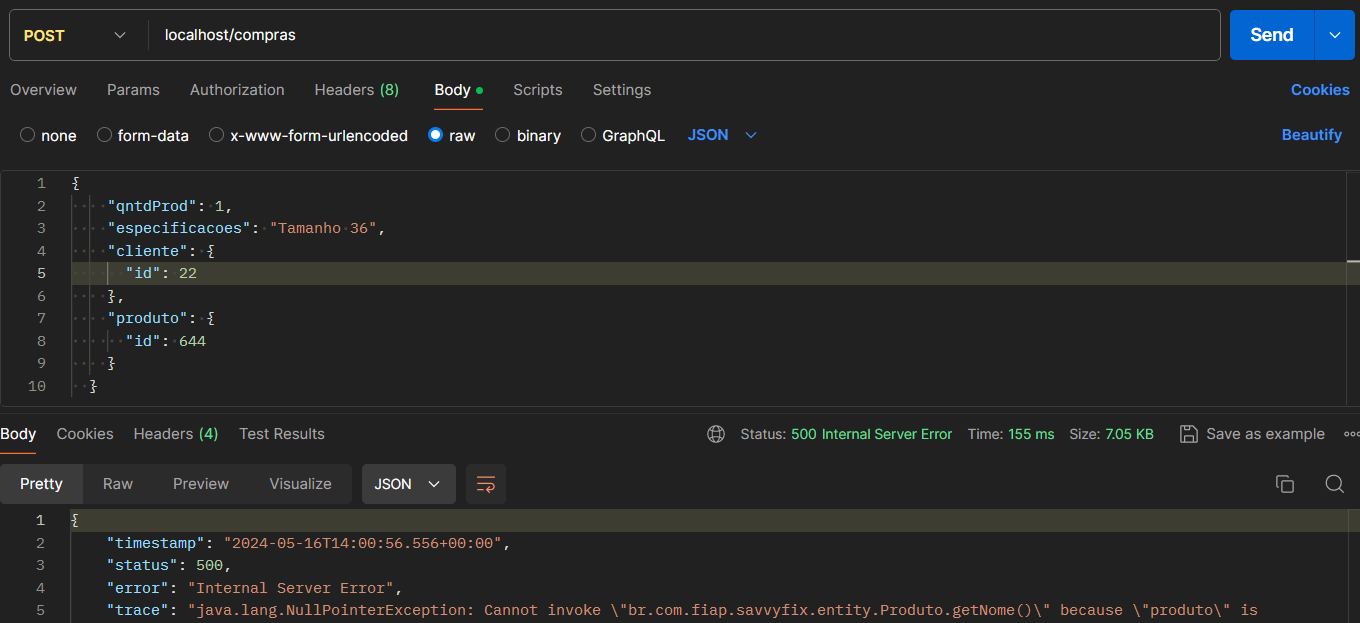
**Compras – POST**

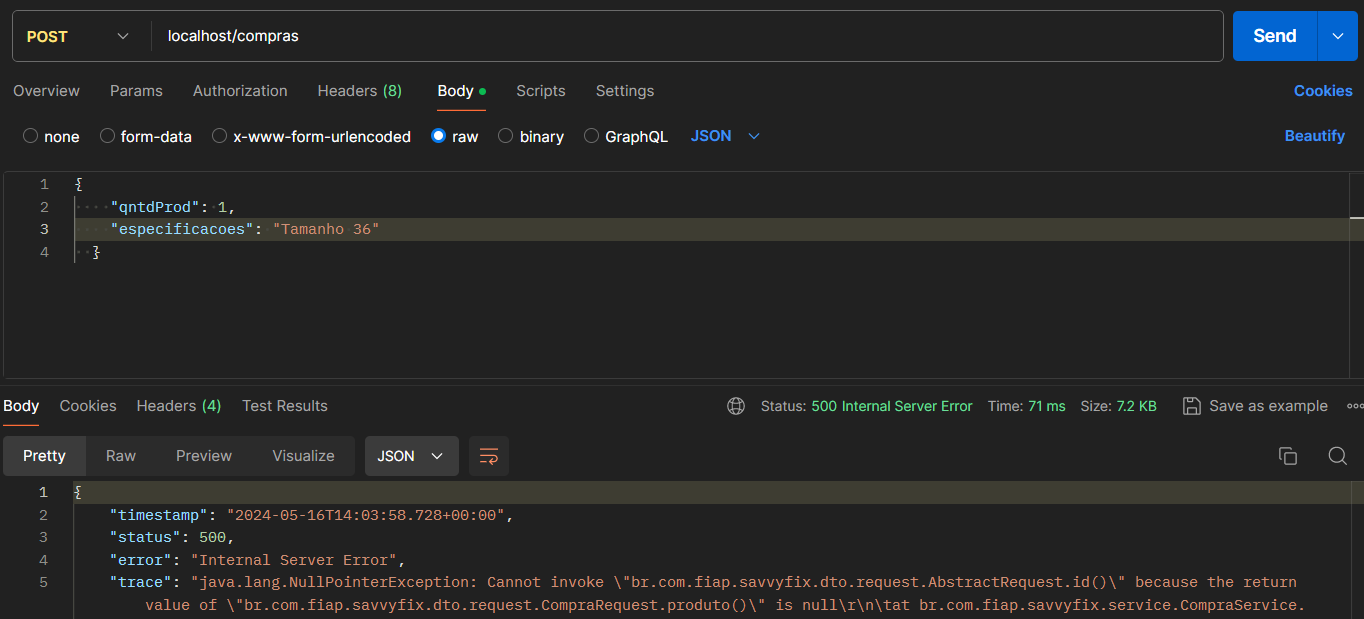
**Endpoint:** localhost/compras

Para salvar os dados é necessário:

1 – Que os Id de cliente e produto sejam válidos e não podem ser nulos







2 – Quantidade e especificações devem ser cadastrados

