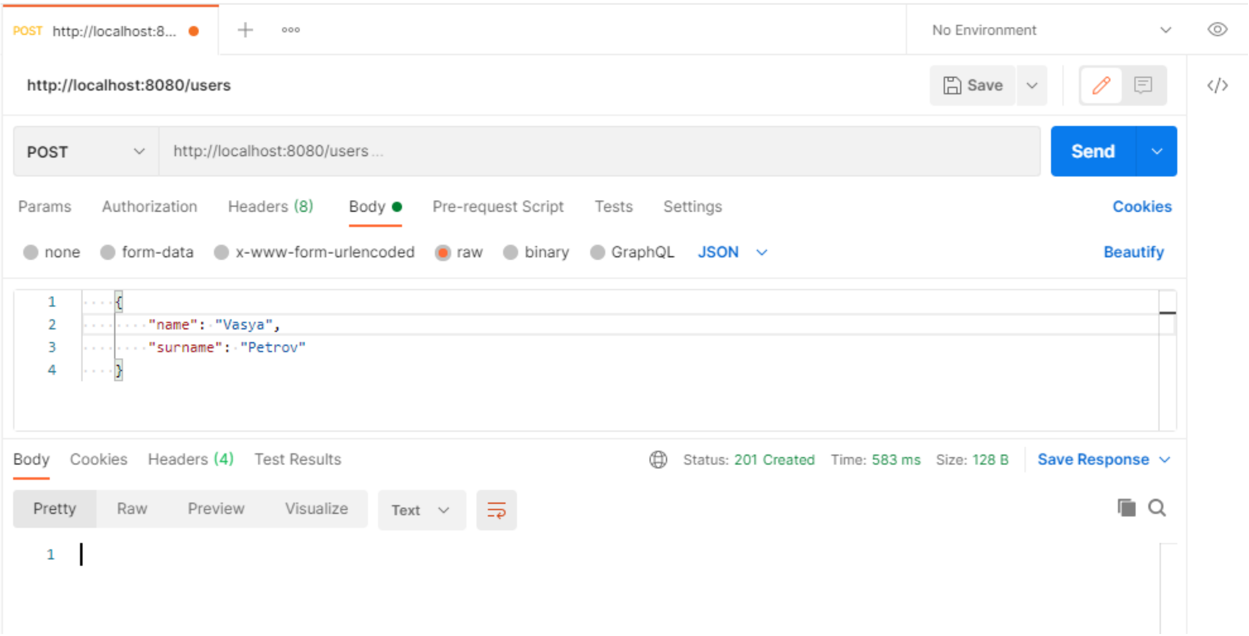
To test the application, I used Postman.

First, let's create a list of phonebook users:

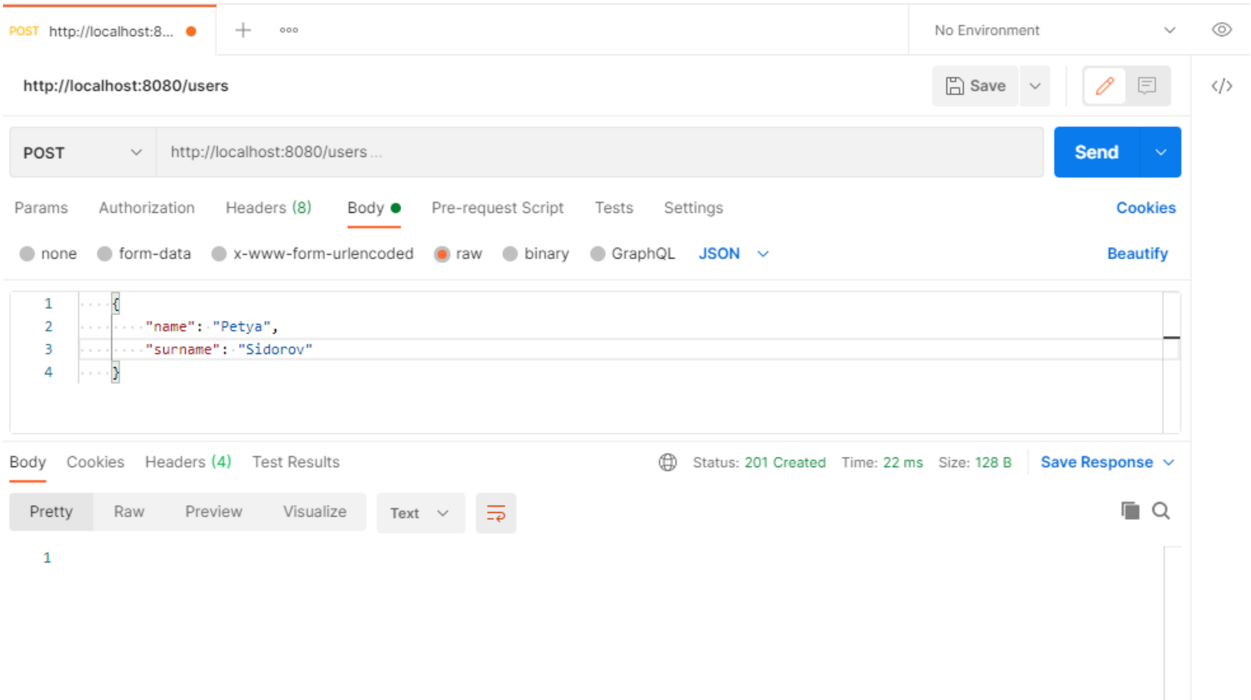


http://localhost:8080/users

POST http://localhost:8080/users

```
{
    "name": "Vasya",
    "surname": "Petrov"
}
```

Status: 201 Created Time: 583 ms Size: 128 B

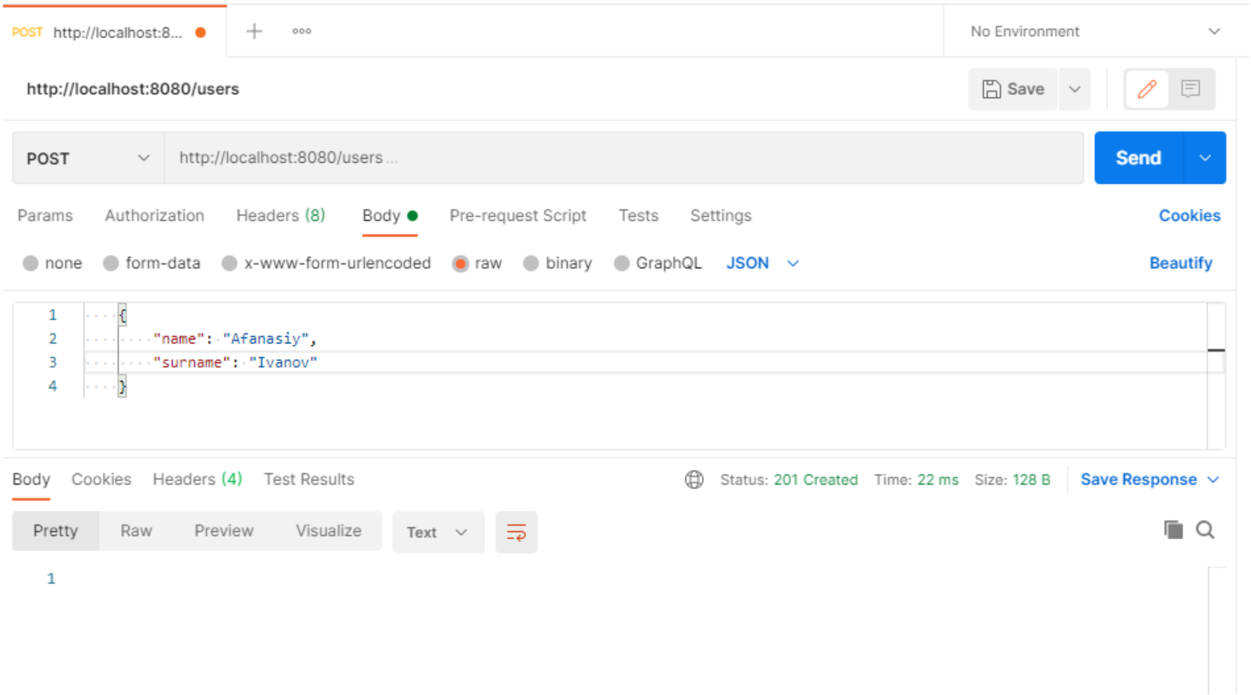


http://localhost:8080/users

POST http://localhost:8080/users

```
{
    "name": "Petya",
    "surname": "Sidorov"
}
```

Status: 201 Created Time: 22 ms Size: 128 B



http://localhost:8080/users

POST http://localhost:8080/users ...

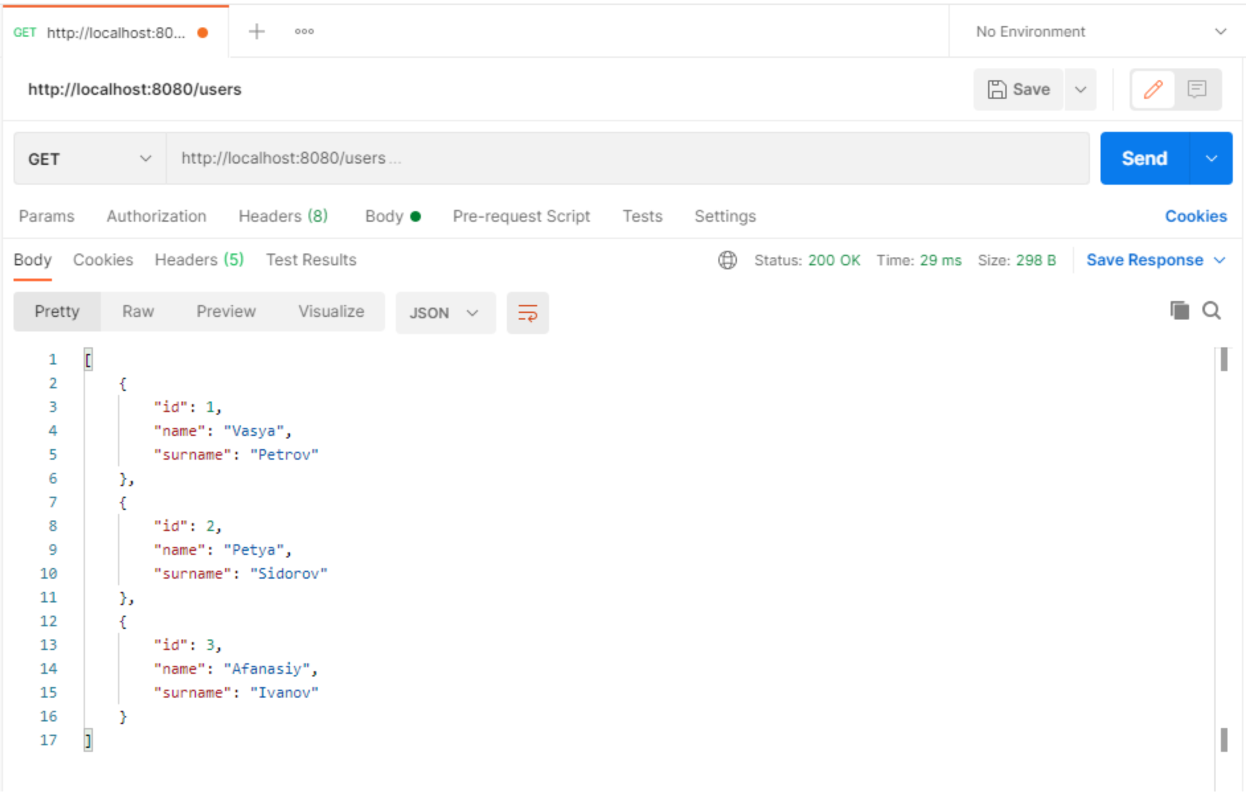
```
{
    "name": "Afanasiy",
    "surname": "Ivanov"
}
```

Status: 201 Created Time: 22 ms Size: 128 B

Text

```
1
```

Check that all are added:



http://localhost:8080/users

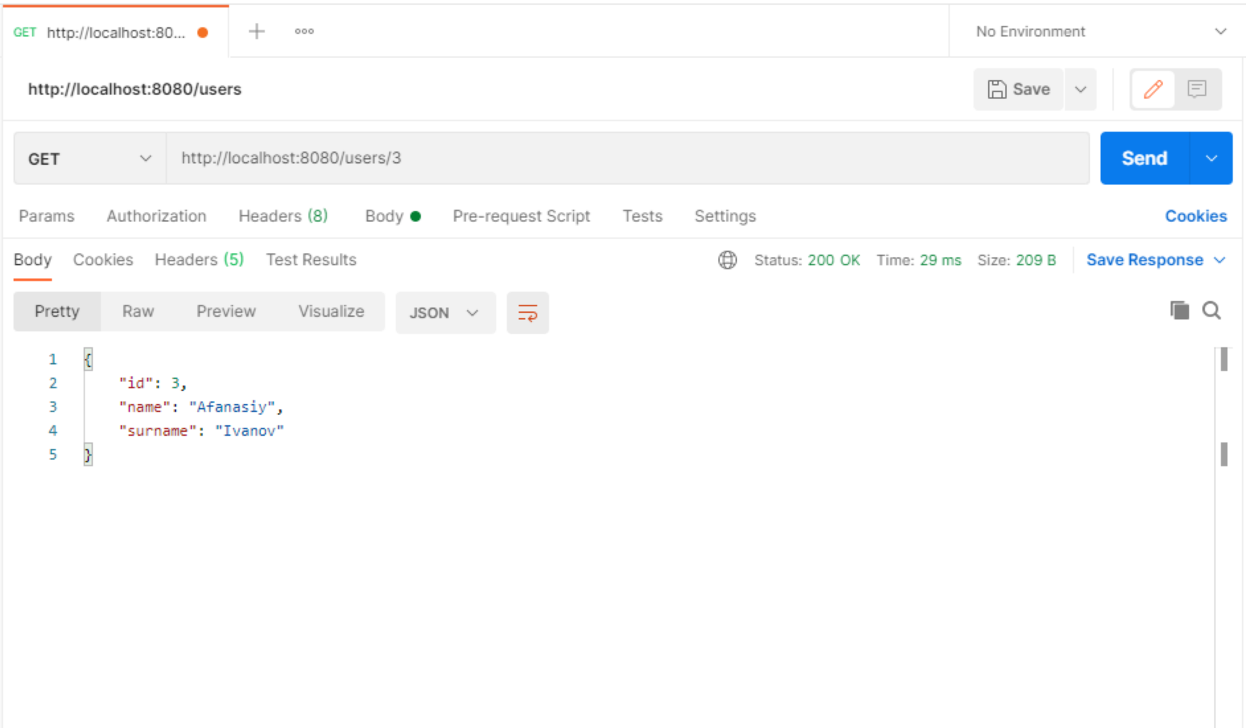
GET http://localhost:8080/users ...

Status: 200 OK Time: 29 ms Size: 298 B

JSON

```
[
    {
        "id": 1,
        "name": "Vasya",
        "surname": "Petrov"
    },
    {
        "id": 2,
        "name": "Petya",
        "surname": "Sidorov"
    },
    {
        "id": 3,
        "name": "Afanasiy",
        "surname": "Ivanov"
    }
]
```

Let's look at user ID 3:



http://localhost:8080/users

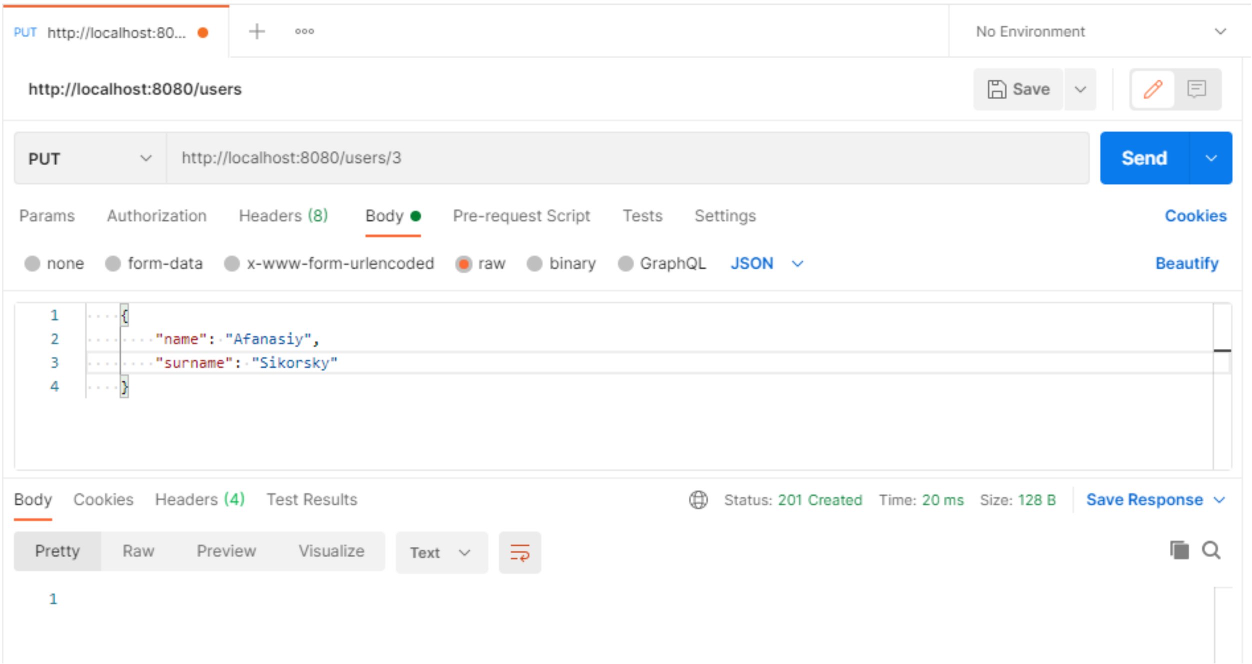
GET http://localhost:8080/users/3 Send

Status: 200 OK Time: 29 ms Size: 209 B

JSON

```
{
    "id": 3,
    "name": "Afanasiy",
    "surname": "Ivanov"
}
```

Let's change his last name:



http://localhost:8080/users

PUT http://localhost:8080/users/3 Send

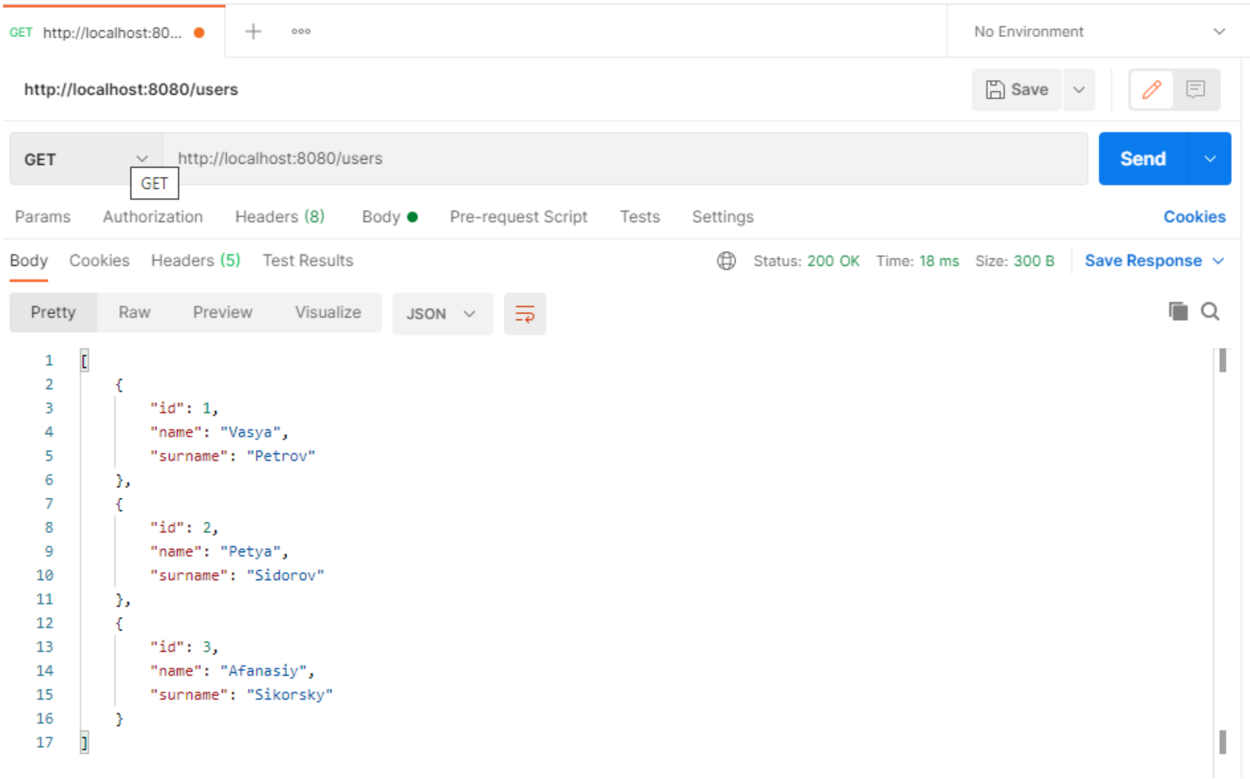
○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
{
    "name": "Afanasiy",
    "surname": "Sikorsky"
}
```

Status: 201 Created Time: 20 ms Size: 128 B

Text

Checking the changes ...



http://localhost:8080/users

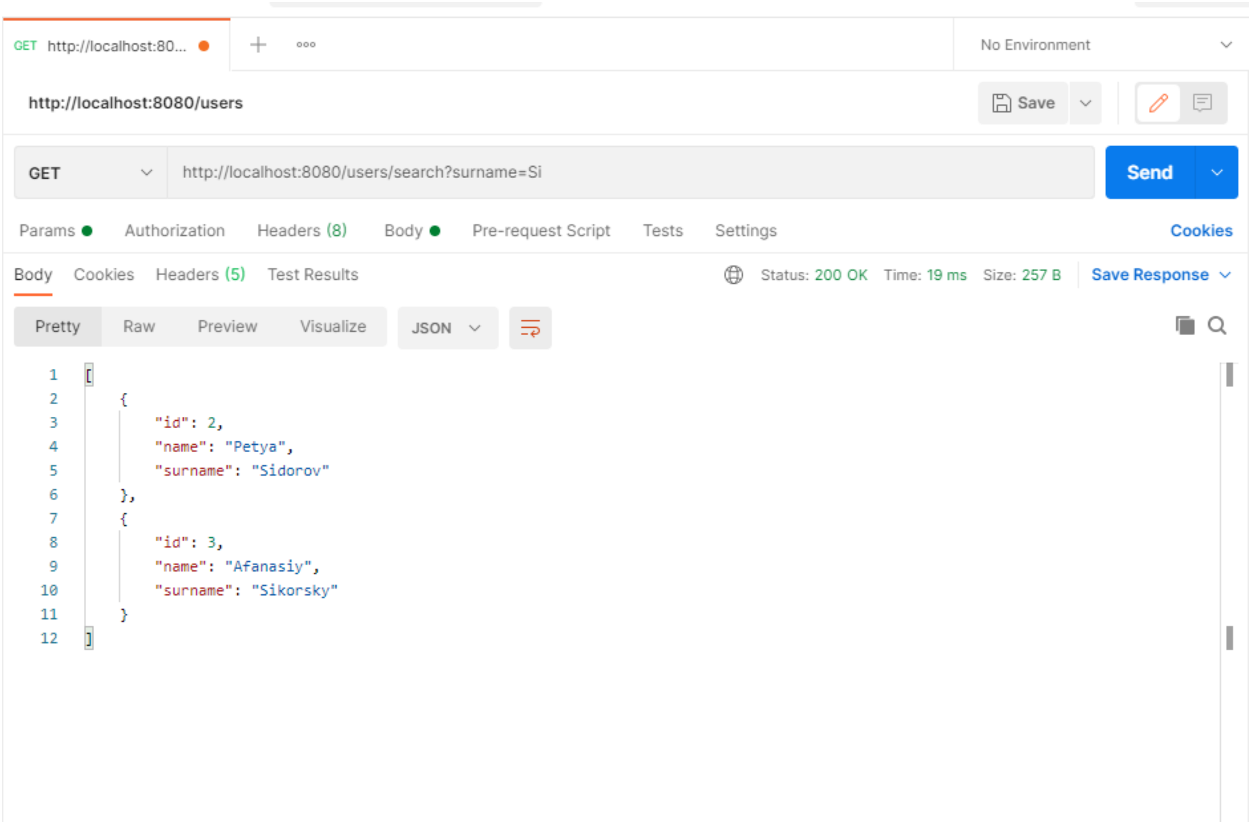
GET http://localhost:8080/users

Status: 200 OK Time: 18 ms Size: 300 B

```
[
    {
        "id": 1,
        "name": "Vasya",
        "surname": "Petrov"
    },
    {
        "id": 2,
        "name": "Petya",
        "surname": "Sidorov"
    },
    {
        "id": 3,
        "name": "Afanasiy",
        "surname": "Sikorsky"
    }
]
```

It worked!

Now let's try to search for users by part of the last name:



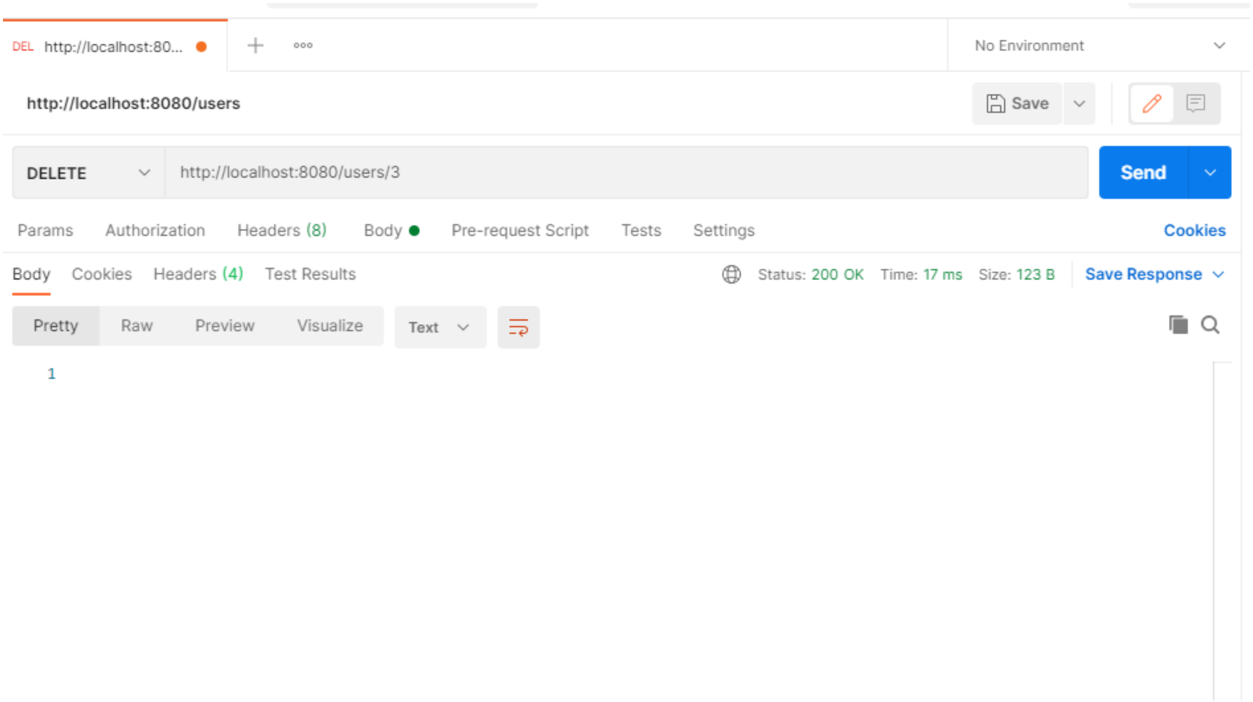
http://localhost:8080/users

GET http://localhost:8080/users/search?surname=Si

Status: 200 OK Time: 19 ms Size: 257 B

```
[
    {
        "id": 2,
        "name": "Petya",
        "surname": "Sidorov"
    },
    {
        "id": 3,
        "name": "Afanasiy",
        "surname": "Sikorsky"
    }
]
```

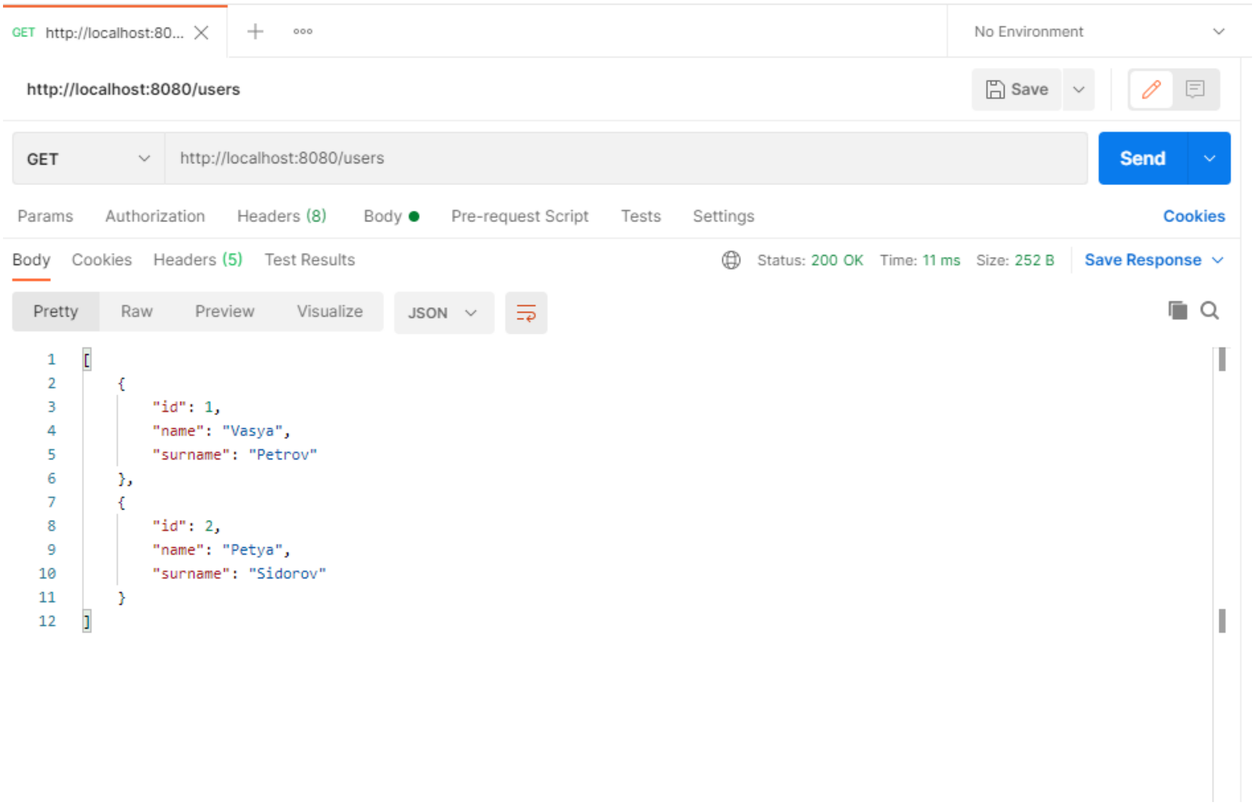
And finally, remove the user:



DELETE http://localhost:8080/users/3

Status: 200 OK Time: 17 ms Size: 123 B

Checking ...:



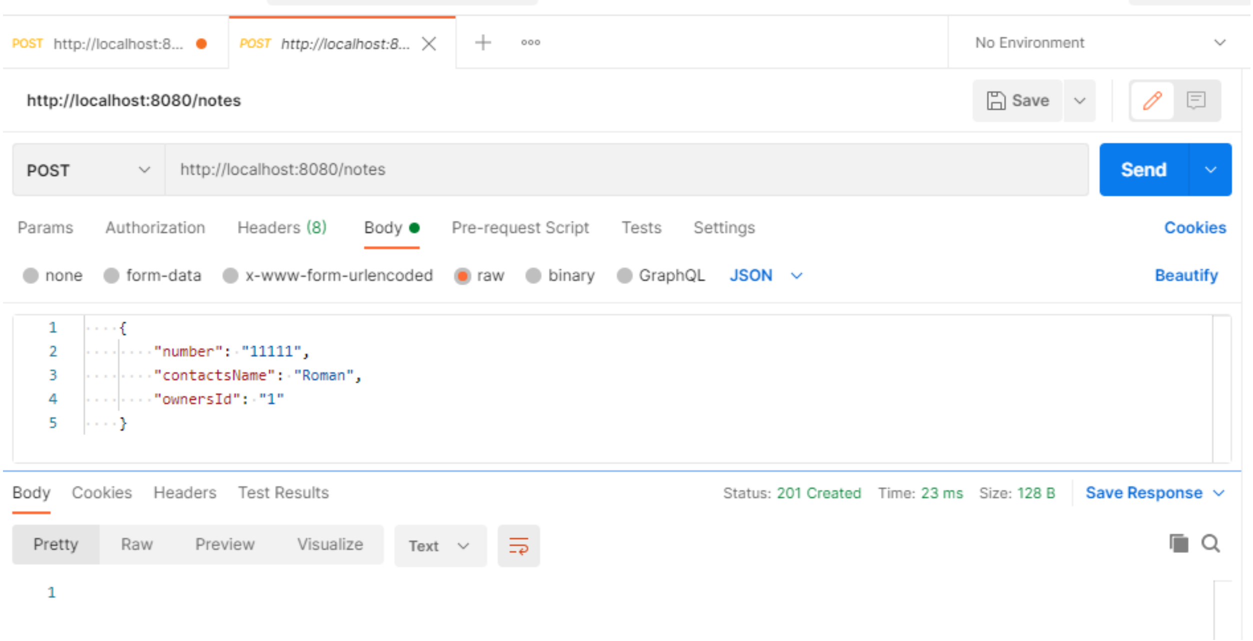
GET http://localhost:8080/users

Status: 200 OK Time: 11 ms Size: 252 B

```
[
    {
        "id": 1,
        "name": "Vasya",
        "surname": "Petrov"
    },
    {
        "id": 2,
        "name": "Petya",
        "surname": "Sidorov"
    }
]
```

It works.

Now add entries to the phone book:

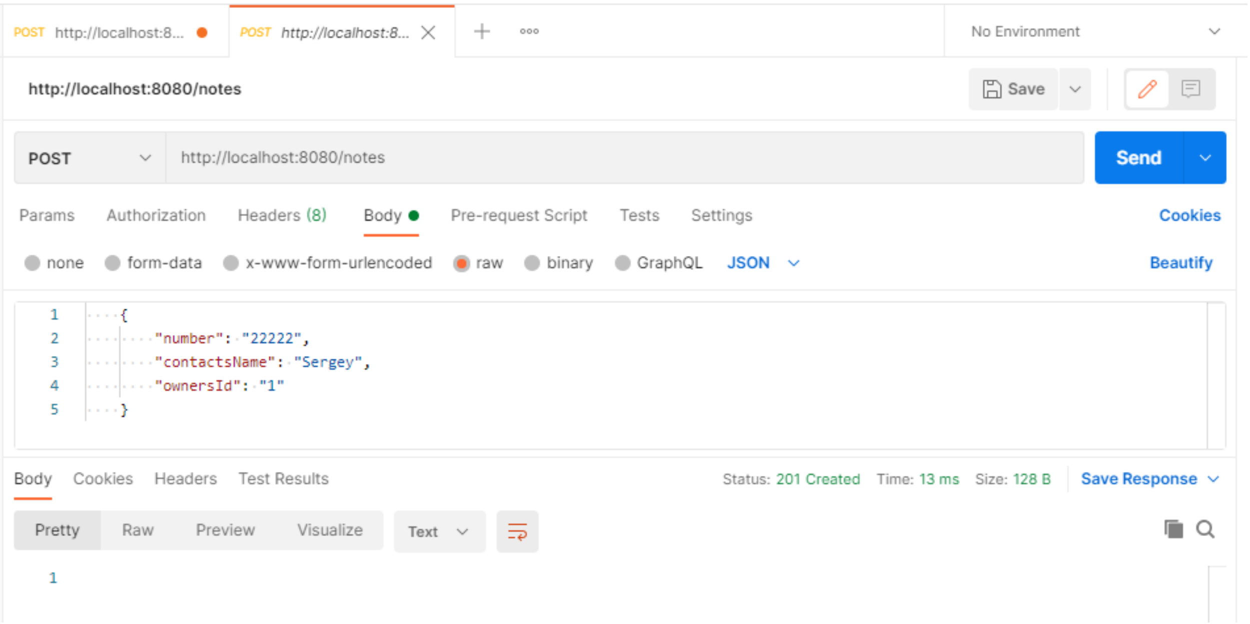


http://localhost:8080/notes

POST http://localhost:8080/notes

```
{
    "number": "11111",
    "contactsName": "Roman",
    "ownersId": "1"
}
```

Status: 201 Created Time: 23 ms Size: 128 B

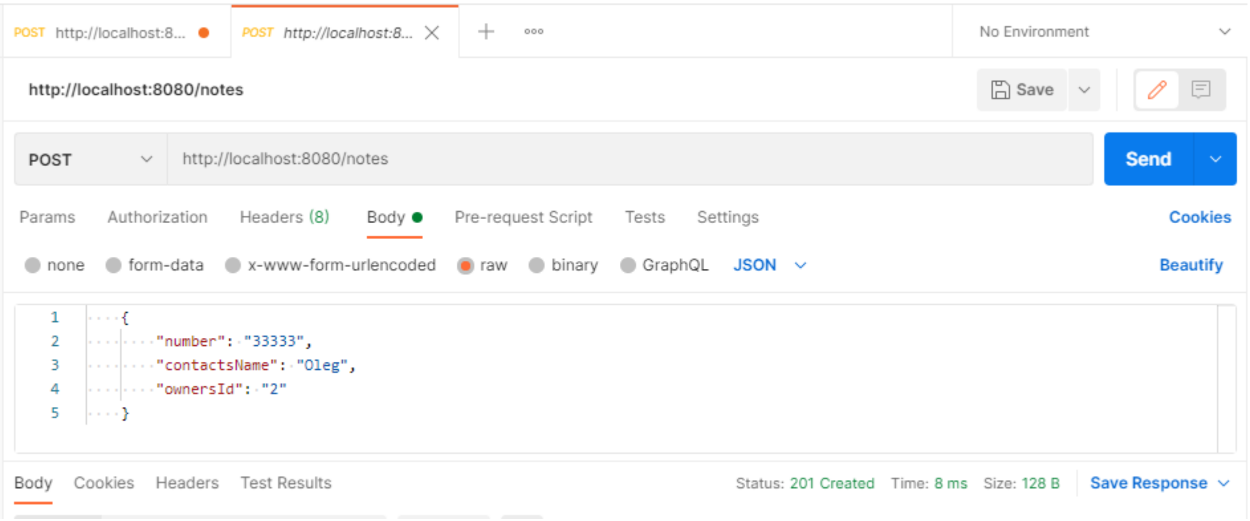


http://localhost:8080/notes

POST http://localhost:8080/notes

```
{
    "number": "22222",
    "contactsName": "Sergey",
    "ownersId": "1"
}
```

Status: 201 Created Time: 13 ms Size: 128 B



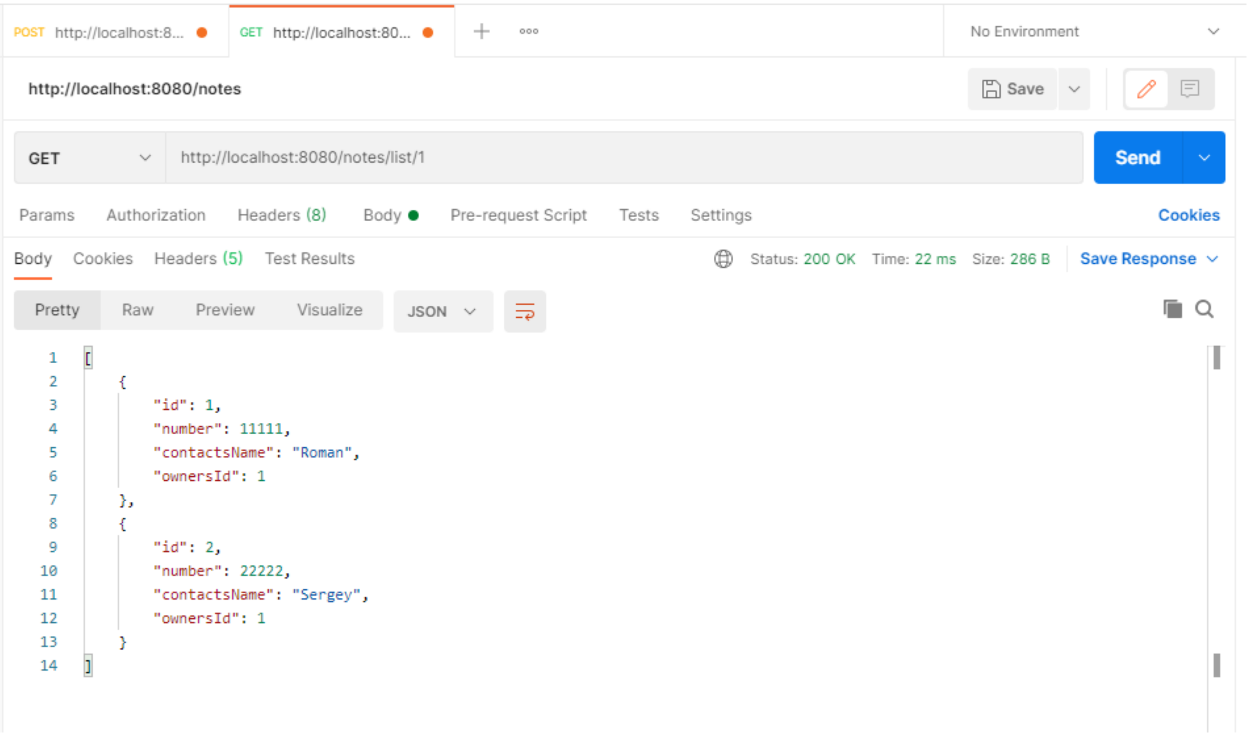
http://localhost:8080/notes

POST http://localhost:8080/notes

```
{
    "number": "33333",
    "contactsName": "Oleg",
    "ownersId": "2"
}
```

Status: 201 Created Time: 8 ms Size: 128 B

Let's see the posts of one of the users:



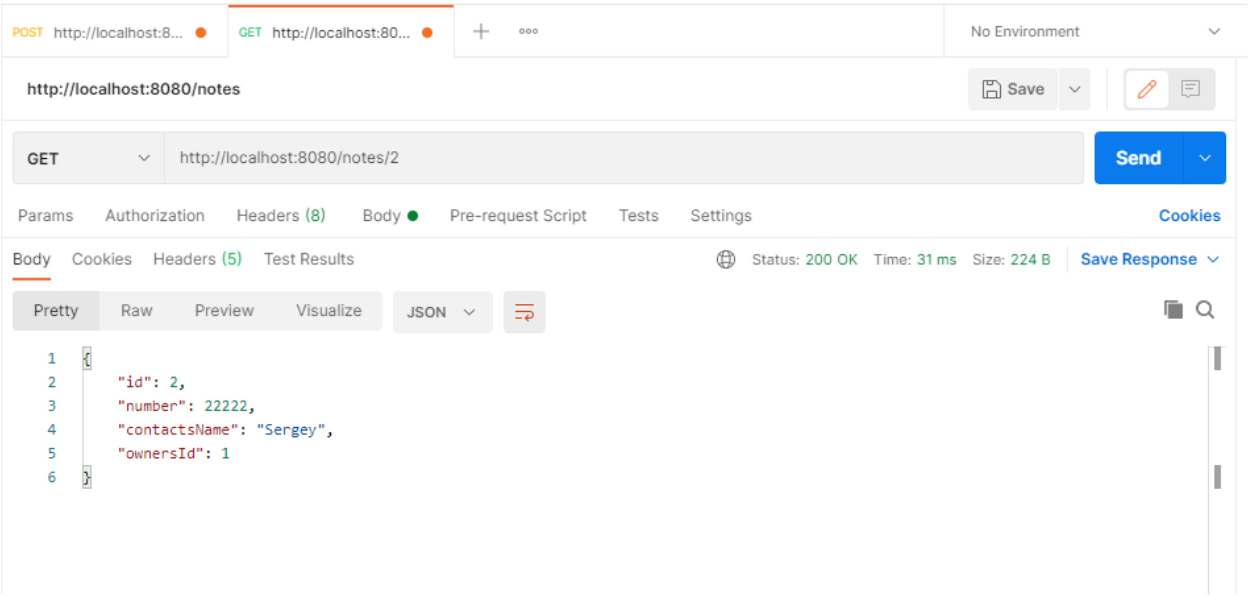
http://localhost:8080/notes

GET http://localhost:8080/notes/list/1

Status: 200 OK Time: 22 ms Size: 286 B

```
[
    {
        "id": 1,
        "number": 11111,
        "contactsName": "Roman",
        "ownersId": 1
    },
    {
        "id": 2,
        "number": 22222,
        "contactsName": "Sergey",
        "ownersId": 1
    }
]
```

Let's see one entry:



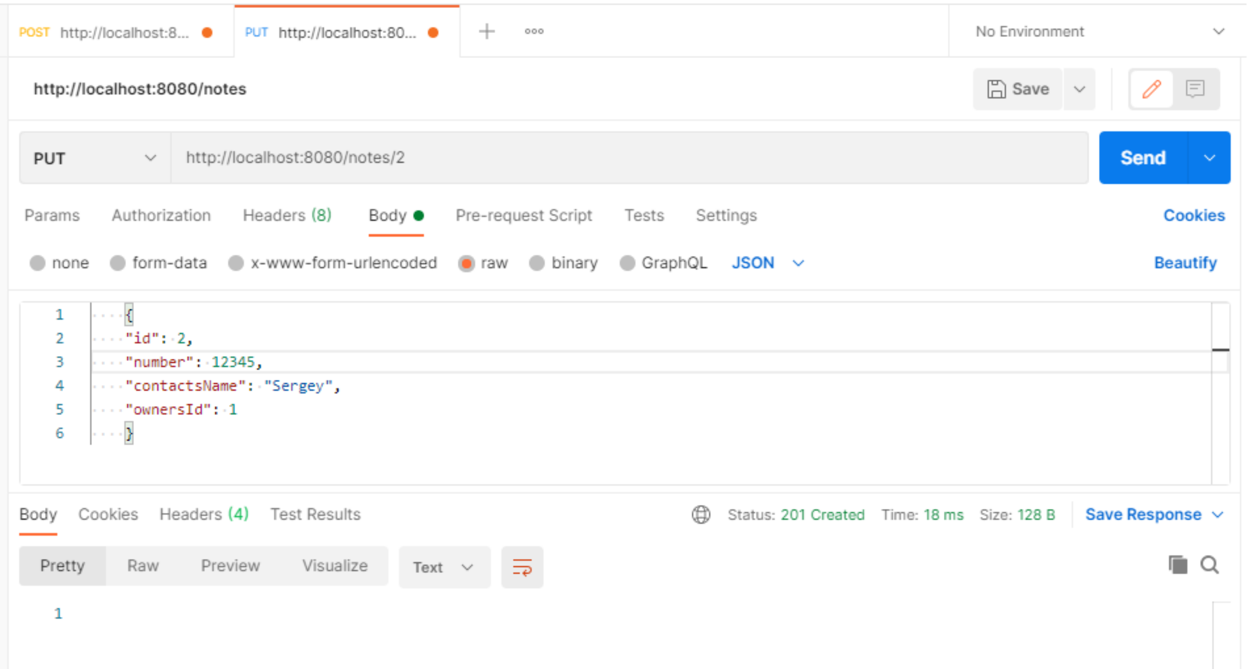
http://localhost:8080/notes

GET http://localhost:8080/notes/2

Status: 200 OK Time: 31 ms Size: 224 B

```
{
    "id": 2,
    "number": 22222,
    "contactsName": "Sergey",
    "ownersId": 1
}
```

Updating:



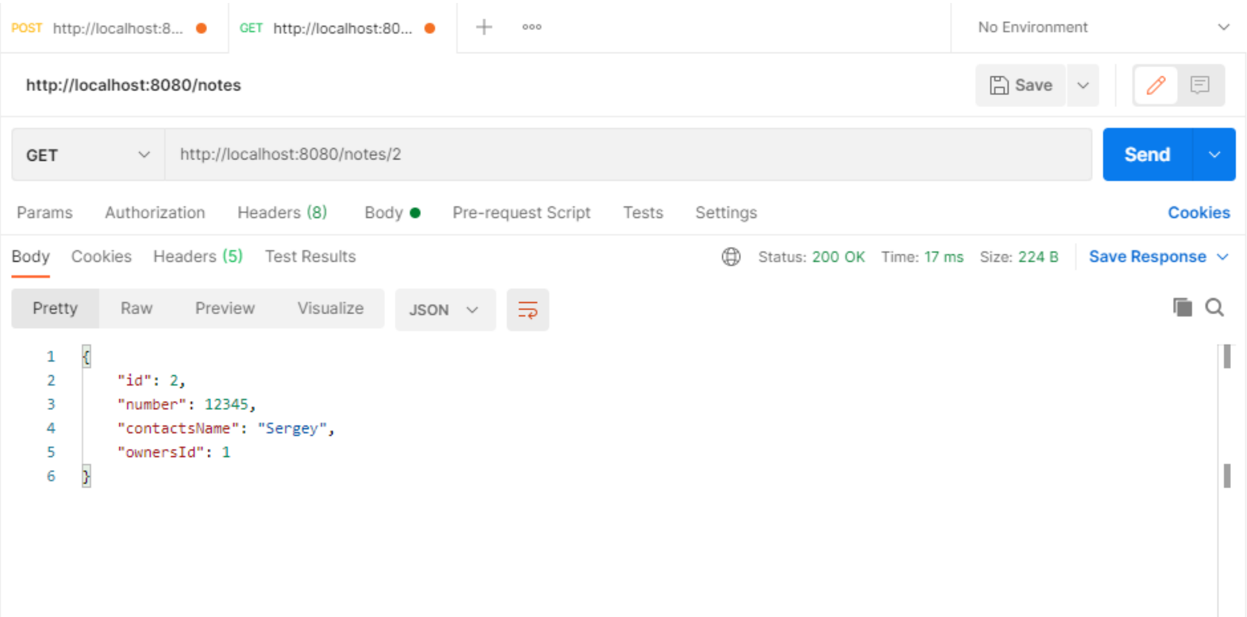
http://localhost:8080/notes

PUT http://localhost:8080/notes/2

```
{
"id": 2,
"number": 12345,
"contactsName": "Sergey",
"ownersId": 1
}
```

Status: 201 Created Time: 18 ms Size: 128 B

Checking the result:



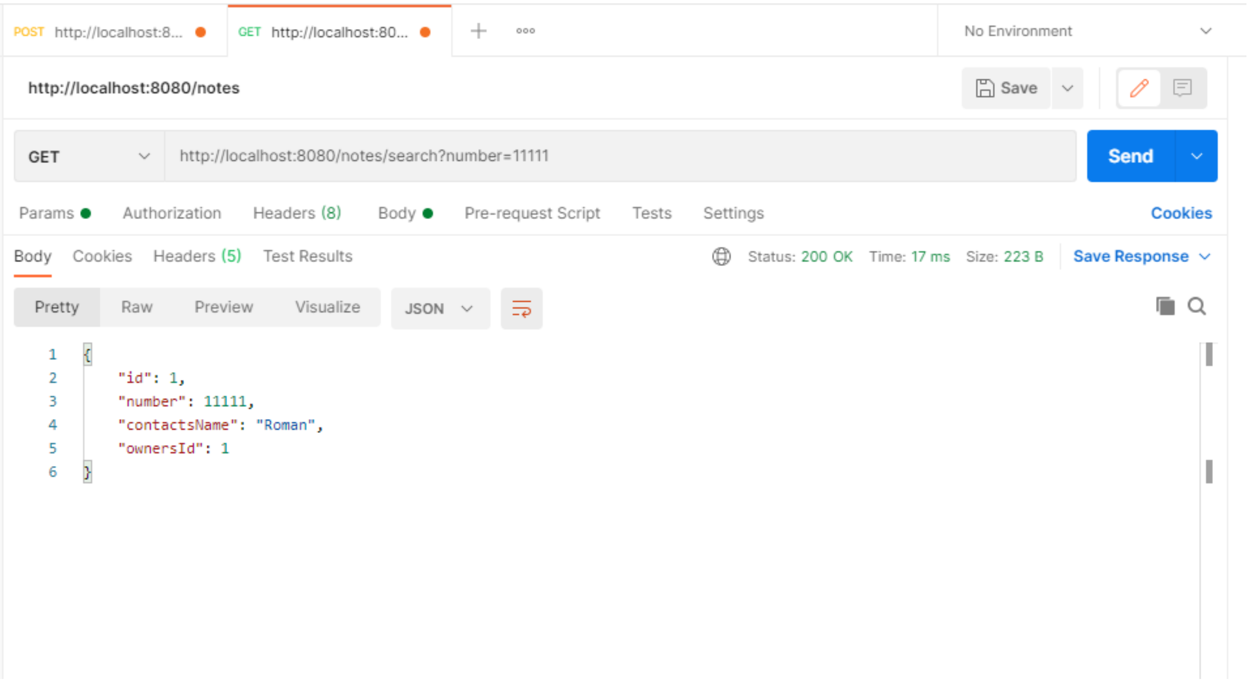
http://localhost:8080/notes

GET http://localhost:8080/notes/2

Status: 200 OK Time: 17 ms Size: 224 B

```
{
    "id": 2,
    "number": 12345,
    "contactsName": "Sergey",
    "ownersId": 1
}
```

Looking for an entry by phone number:



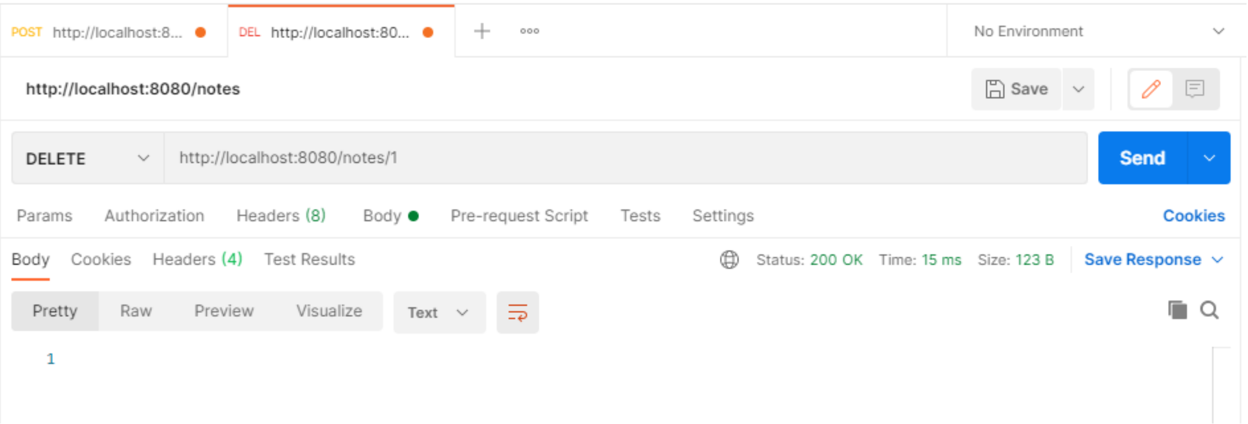
http://localhost:8080/notes

GET http://localhost:8080/notes/search?number=11111

Status: 200 OK Time: 17 ms Size: 223 B

```
{
    "id": 1,
    "number": 11111,
    "contactsName": "Roman",
    "ownersId": 1
}
```

And delete it:

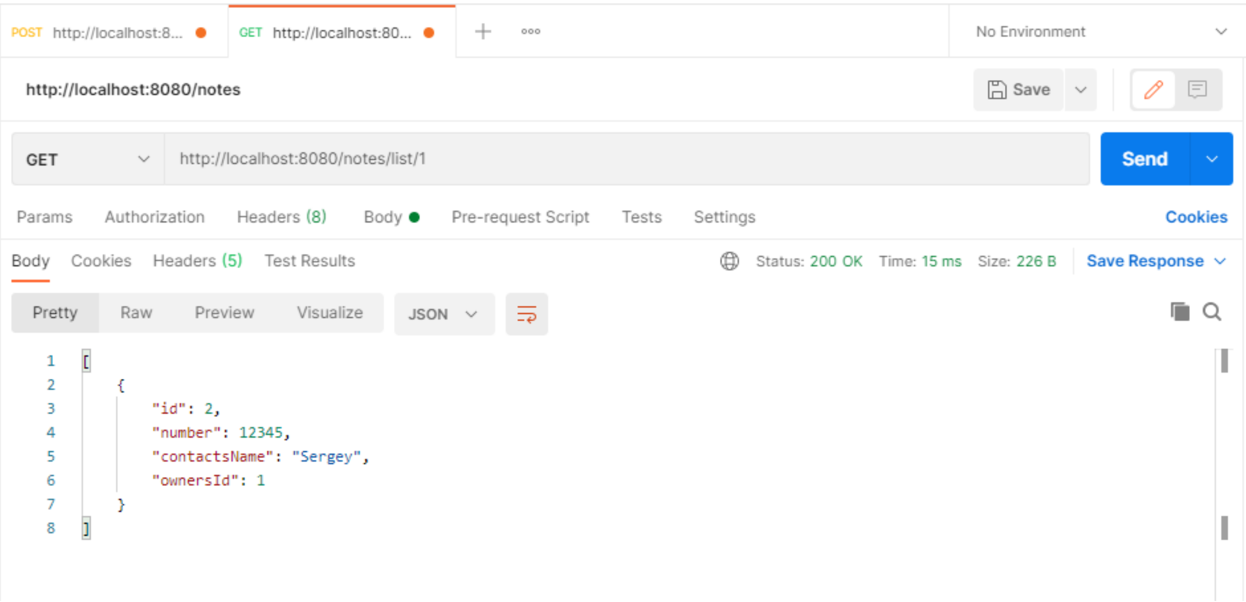


http://localhost:8080/notes

DELETE http://localhost:8080/notes/1

Status: 200 OK Time: 15 ms Size: 123 B

Checking the result:



http://localhost:8080/notes/list/1

Status: 200 OK Time: 15 ms Size: 226 B

```
[
    {
        "id": 2,
        "number": 12345,
        "contactsName": "Sergey",
        "ownersId": 1
    }
]
```