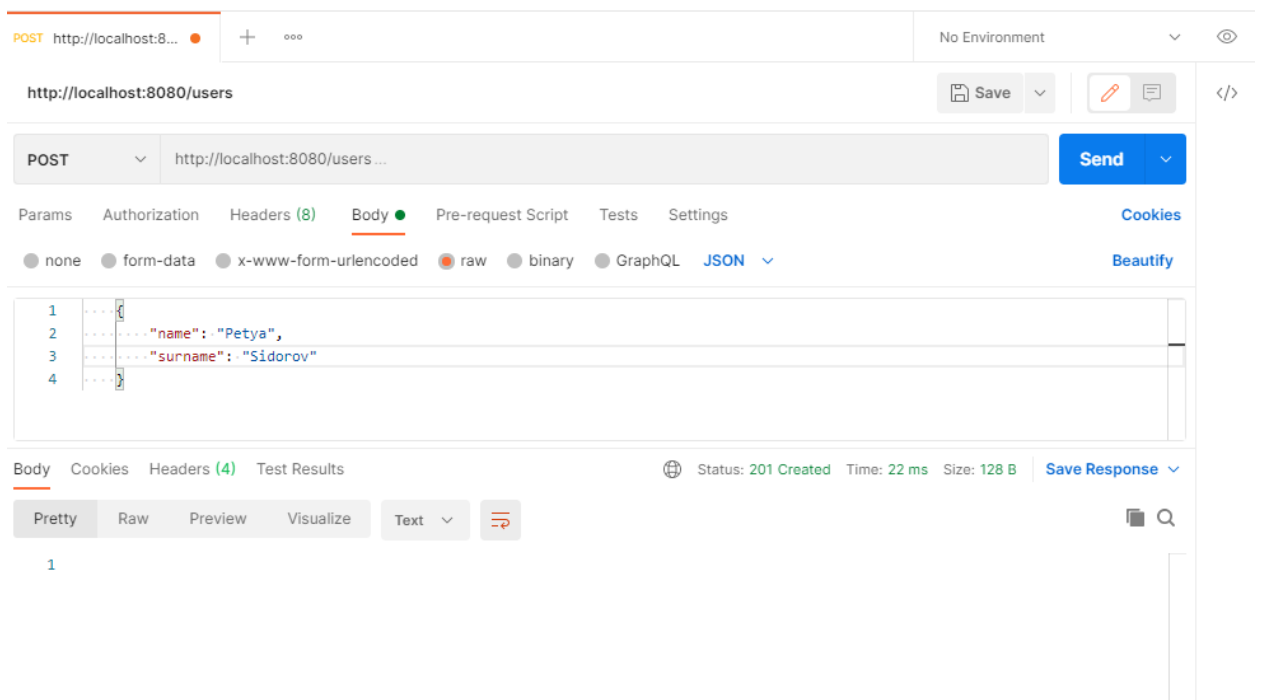
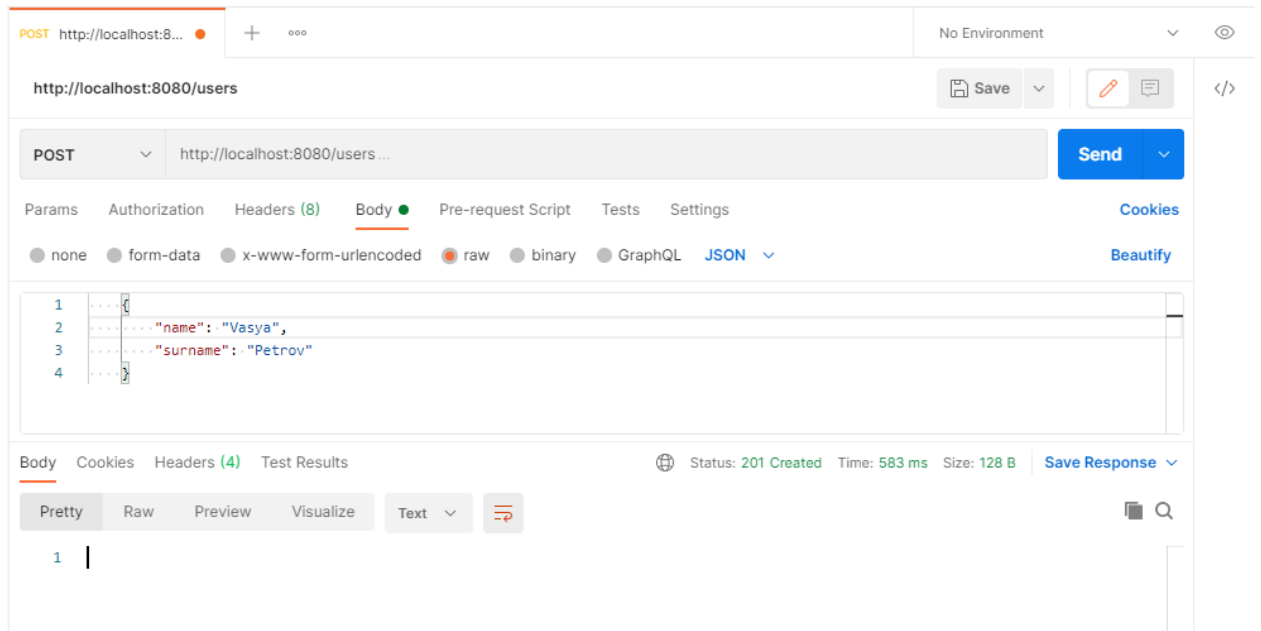


To test the application, I used Postman.

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



POST http://localhost:8080/users No Environment

Save

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Beautify

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

```
1 {
2   "name": "Afanasiy",
3   "surname": "Ivanov"
4 }
```

Body Cookies Headers (4) Test Results Status: 201 Created Time: 22 ms Size: 128 B Save Response

Pretty Raw Preview Visualize Text

1

Check that all are added:

GET http://localhost:8080/users No Environment

Save

Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (5) Test Results Status: 200 OK Time: 29 ms Size: 298 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "id": 1,
4     "name": "Vasya",
5     "surname": "Petrov"
6   },
7   {
8     "id": 2,
9     "name": "Petya",
10    "surname": "Sidorov"
11  },
12  {
13    "id": 3,
14    "name": "Afanasiy",
15    "surname": "Ivanov"
16  }
17 ]
```

Let's look at user ID 3:

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

Save Send

Params Authorization Headers (8) Body ● Pre-request Script Tests Settings Cookies

Body Cookies Headers (5) Test Results Status: 200 OK Time: 29 ms Size: 209 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 3,
3   "name": "Afanasiy",
4   "surname": "Ivanov"
5 }
```

Let's change his last name:

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

Save Send

Params Authorization Headers (8) Body ● Pre-request Script Tests Settings Cookies

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

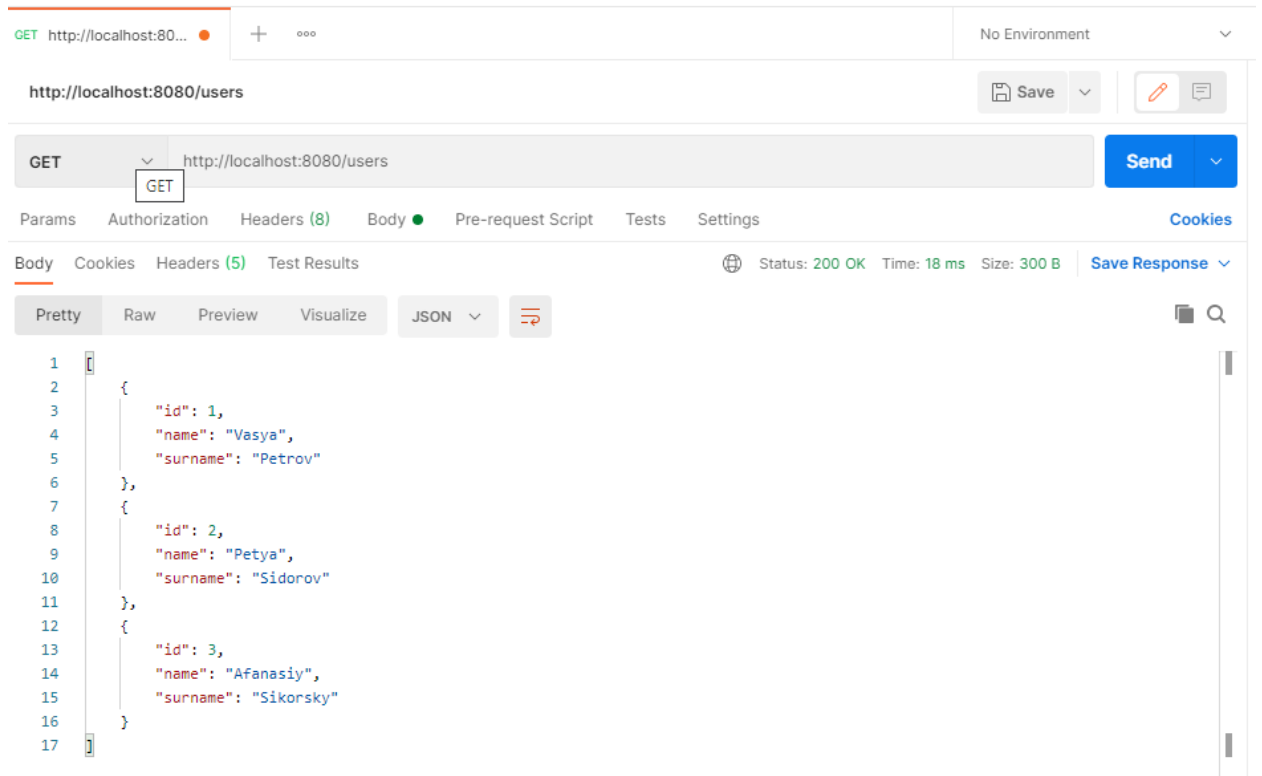
```
1 {
2   "name": "Afanasiy",
3   "surname": "Sikorsky"
4 }
```

Body Cookies Headers (4) Test Results Status: 201 Created Time: 20 ms Size: 128 B Save Response

Pretty Raw Preview Visualize Text

```
1
```

Checking the changes ...



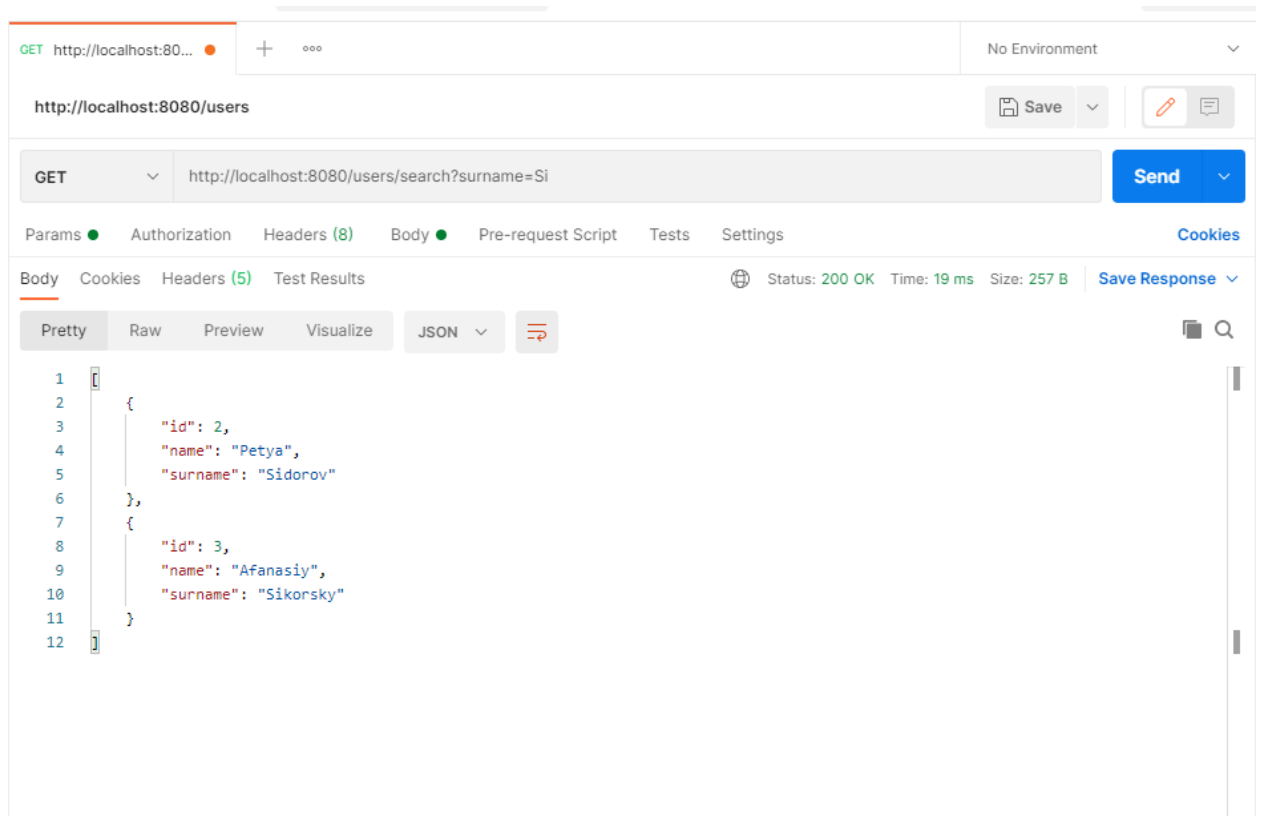
The screenshot shows a REST client interface with the following details:

- Request:** GET `http://localhost:8080/users`
- Status:** 200 OK, Time: 18 ms, Size: 300 B
- Response Body (JSON):**

```
[
  {
    "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:



The screenshot shows a REST client interface with the following details:

- Request:** GET `http://localhost:8080/users/search?surname=Si`
- Status:** 200 OK, Time: 19 ms, Size: 257 B
- Response Body (JSON):**

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

And finally, remove the user:

The screenshot shows the Postman interface with a DELETE request configured. The URL bar at the top shows 'http://localhost:8080/users/3'. The request method is set to 'DELETE'. The 'Send' button is visible on the right. Below the request bar, the 'Body' tab is selected, and the response status is '200 OK' with a time of '17 ms' and size of '123 B'. The response body is empty, indicated by a '1' on the first line.

Checking ...:

The screenshot shows the Postman interface with a GET request configured. The URL bar at the top shows 'http://localhost:8080/users'. The request method is set to 'GET'. The 'Send' button is visible on the right. Below the request bar, the 'Body' tab is selected, and the response status is '200 OK' with a time of '11 ms' and size of '252 B'. The response body is displayed in JSON format, showing an array of two user objects:

```
1 [
2   {
3     "id": 1,
4     "name": "Vasya",
5     "surname": "Petrov"
6   },
7   {
8     "id": 2,
9     "name": "Petya",
10    "surname": "Sidorov"
11  }
12 ]
```

It works.

Now add entries to the phone book:

POST http://localhost:8080/notes

POST http://localhost:8080/notes

No Environment

Save

POST

http://localhost:8080/notes

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

1

2

3

4

5

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

Body

Cookies

Headers

Test Results

Status: 201 Created

Time: 23 ms

Size: 128 B

Save Response

Pretty

Raw

Preview

Visualize

Text

1

POST http://localhost:8080/notes

POST http://localhost:8080/notes

No Environment

Save

POST

http://localhost:8080/notes

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

1

2

3

4

5

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

Body

Cookies

Headers

Test Results

Status: 201 Created

Time: 13 ms

Size: 128 B

Save Response

Pretty

Raw

Preview

Visualize

Text

1

POST http://localhost:8080/notes

POST http://localhost:8080/notes

No Environment

Save

POST

http://localhost:8080/notes

Send

Params

Authorization

Headers (8)

Body

Pre-request Script

Tests

Settings

Cookies

none

form-data

x-www-form-urlencoded

raw

binary

GraphQL

JSON

Beautify

1

2

3

4

5

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

Body

Cookies

Headers

Test Results

Status: 201 Created

Time: 8 ms

Size: 128 B

Save Response

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

POST http://localhost:8... GET http://localhost:80...

http://localhost:8080/notes

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

Params Authorization Headers (8) Body ● Pre-request Script Tests Settings Cookies

Body Cookies Headers (5) Test Results Status: 200 OK Time: 22 ms Size: 286 B Save Response

```
1 [
2   {
3     "id": 1,
4     "number": 11111,
5     "contactsName": "Roman",
6     "ownersId": 1
7   },
8   {
9     "id": 2,
10    "number": 22222,
11    "contactsName": "Sergey",
12    "ownersId": 1
13  }
14 ]
```

Let's see one entry:

POST http://localhost:8... GET http://localhost:80...

http://localhost:8080/notes

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

Params Authorization Headers (8) Body ● Pre-request Script Tests Settings Cookies

Body Cookies Headers (5) Test Results Status: 200 OK Time: 31 ms Size: 224 B Save Response

```
1 {
2   "id": 2,
3   "number": 22222,
4   "contactsName": "Sergey",
5   "ownersId": 1
6 }
```

Updating:

The screenshot shows a REST client interface with a PUT request to `http://localhost:8080/notes/2`. The request body is a JSON object: `{ "id": 2, "number": 12345, "contactsName": "Sergey", "ownersId": 1 }`. The status bar indicates a 201 Created response with a time of 18 ms and a size of 128 B.

```
1 {
2   "id": 2,
3   "number": 12345,
4   "contactsName": "Sergey",
5   "ownersId": 1
6 }
```

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

Checking the result:

The screenshot shows a REST client interface with a GET request to `http://localhost:8080/notes`. The response body is a JSON object: `{ "id": 2, "number": 12345, "contactsName": "Sergey", "ownersId": 1 }`. The status bar indicates a 200 OK response with a time of 17 ms and a size of 224 B.

```
1 {
2   "id": 2,
3   "number": 12345,
4   "contactsName": "Sergey",
5   "ownersId": 1
6 }
```

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

Looking for an entry by phone number:

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

Save Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (5) Test Results Status: 200 OK Time: 17 ms Size: 223 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "id": 1,
3   "number": 11111,
4   "contactsName": "Roman",
5   "ownersId": 1
6 }
```

And delete it:

POST http://localhost:8080/notes DEL http://localhost:8080/notes/1

Save Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (4) Test Results Status: 200 OK Time: 15 ms Size: 123 B Save Response

Pretty Raw Preview Visualize Text

```
1
```

Checking the result:

POST http://localhost:8080/notes

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

No Environment

http://localhost:8080/notes

Save

Send

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

ParamsAuthorizationHeaders (8)Body ●Pre-request ScriptTestsSettings

Cookies

BodyCookiesHeaders (5)Test Results

⊕ Status: 200 OKTime: 15 msSize: 226 BSave Response

PrettyRawPreviewVisualizeJSON

```
1 [
2   {
3     "id": 2,
4     "number": 12345,
5     "contactsName": "Sergey",
6     "ownersId": 1
7   }
8 ]
```