https://gorest.co.in/

Задание

1. Получите список пользователей.
Каким запросом получили список пользователей?
Какой статус ответа получили? Что содержится в теле ответа? Сделайте скриншот.

2. Модифицируйте запрос и получите страницу 12. Что содержится в теле ответа? Сделайте скриншот.

3. Какие отличия запроса в п.2 от запроса в п.1?

4. Сделайте post запрос для получения пользователя с пустым телом запроса. Какой ответ пришел и почему? Сделайте скриншот.

5. Напишите запрос для создания пользователя, используя метод raw => json. Какой ответ пришел? Сделайте скриншот.

6. Напишите запрос, возвращающий в ответе запись о созданном пользователе. Сделайте скриншот.

7. Создайте еще несколько пользователей с одинаковыми значениями "name", но разными email. Получите список всех только что созданных пользователей.
Сделайте скриншот.
Какими ещё get запросами можно получить запись о пользователе?

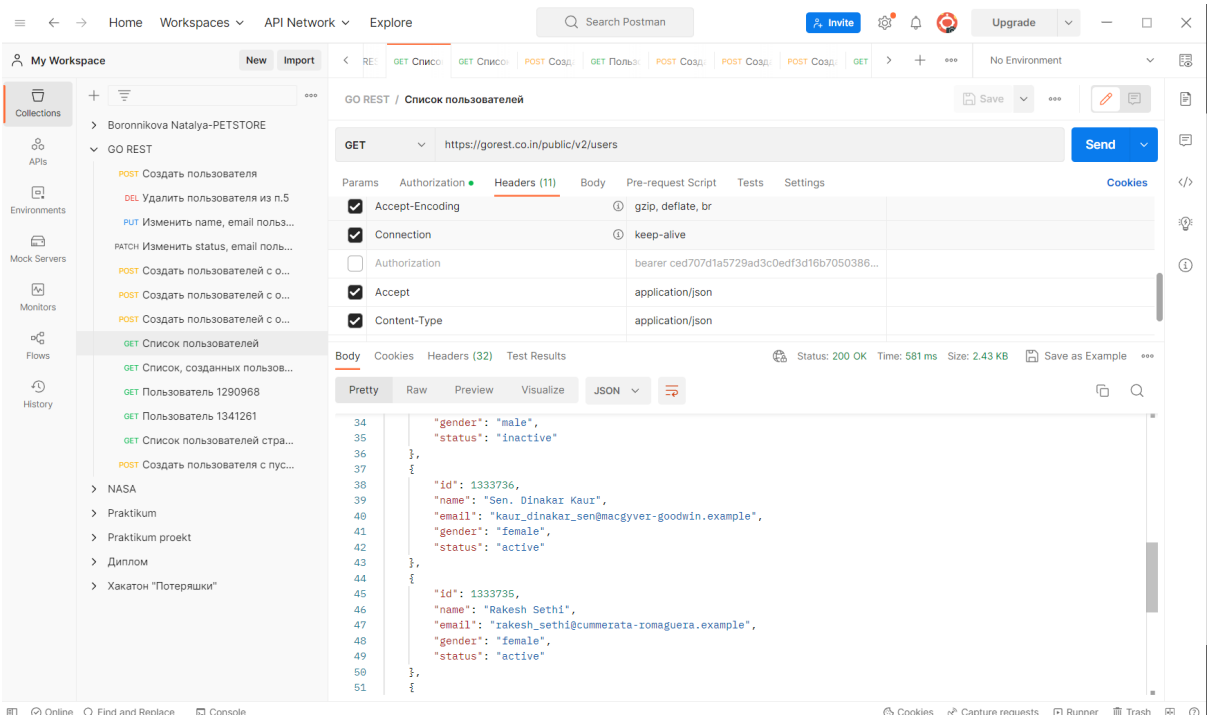
8. Напишите запрос с методом put на изменение email и name на другое значение. Проверьте изменение get запросом. Сделайте скриншот.

9. Чем отличаются запросы put и patch?
Напишите запрос с методом patch для корректировки полей status и email. Сделайте скриншот.
В чем отличие полученного результата, от результата запроса из п.8?

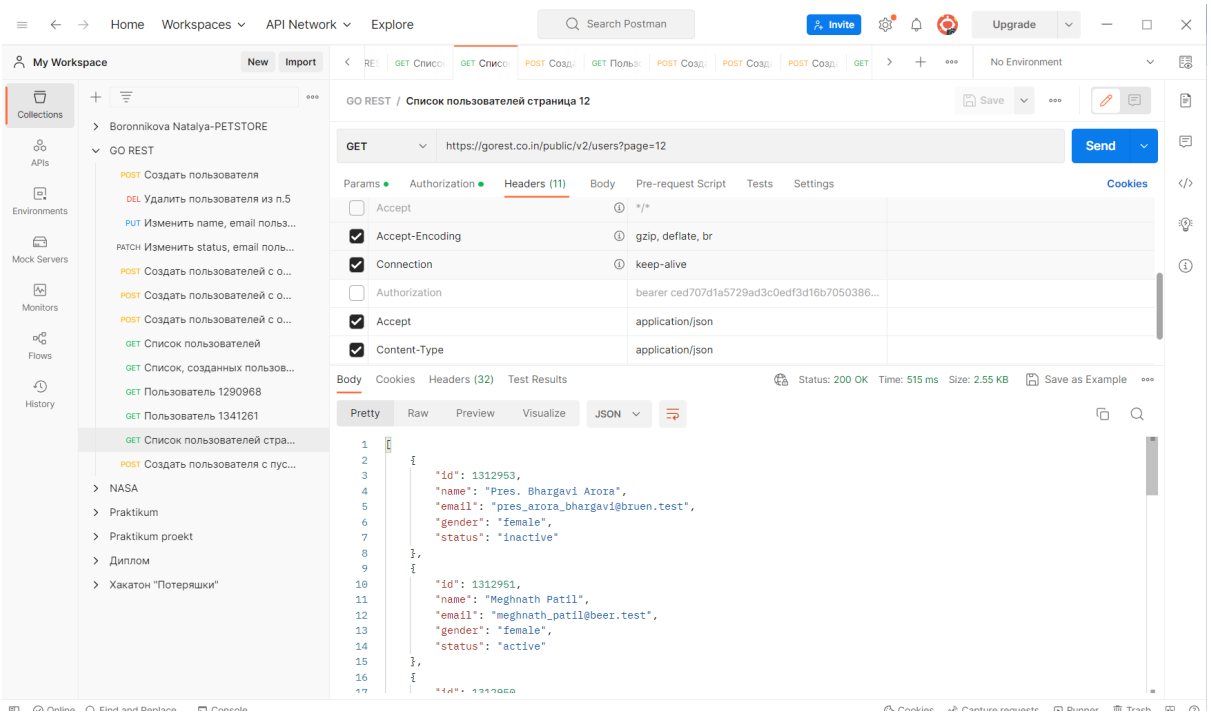
10. Удалите пользователя, созданного в п.5. Что содержится в теле ответа? Сделайте скриншот.
Выполните get запрос удаленного пользователя. Какой ответ пришел и почему? Сделайте скриншот.

11. Удалите всех ранее созданных пользователей.

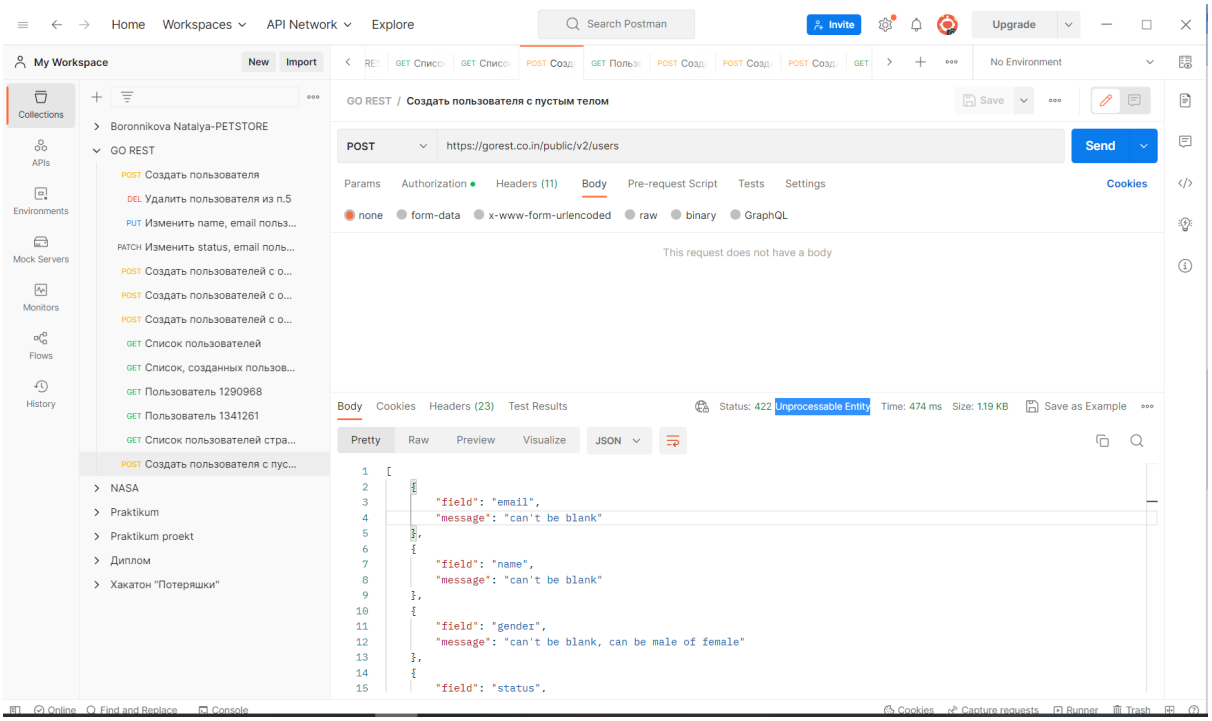
1. Список пользователей получен с помощью метода GET.
   Статус ответа 200 OK.
   Тело ответа состоит из массива объектов, в объекте содержится ин-я об id, name, email, status, gender. Формат данных json.



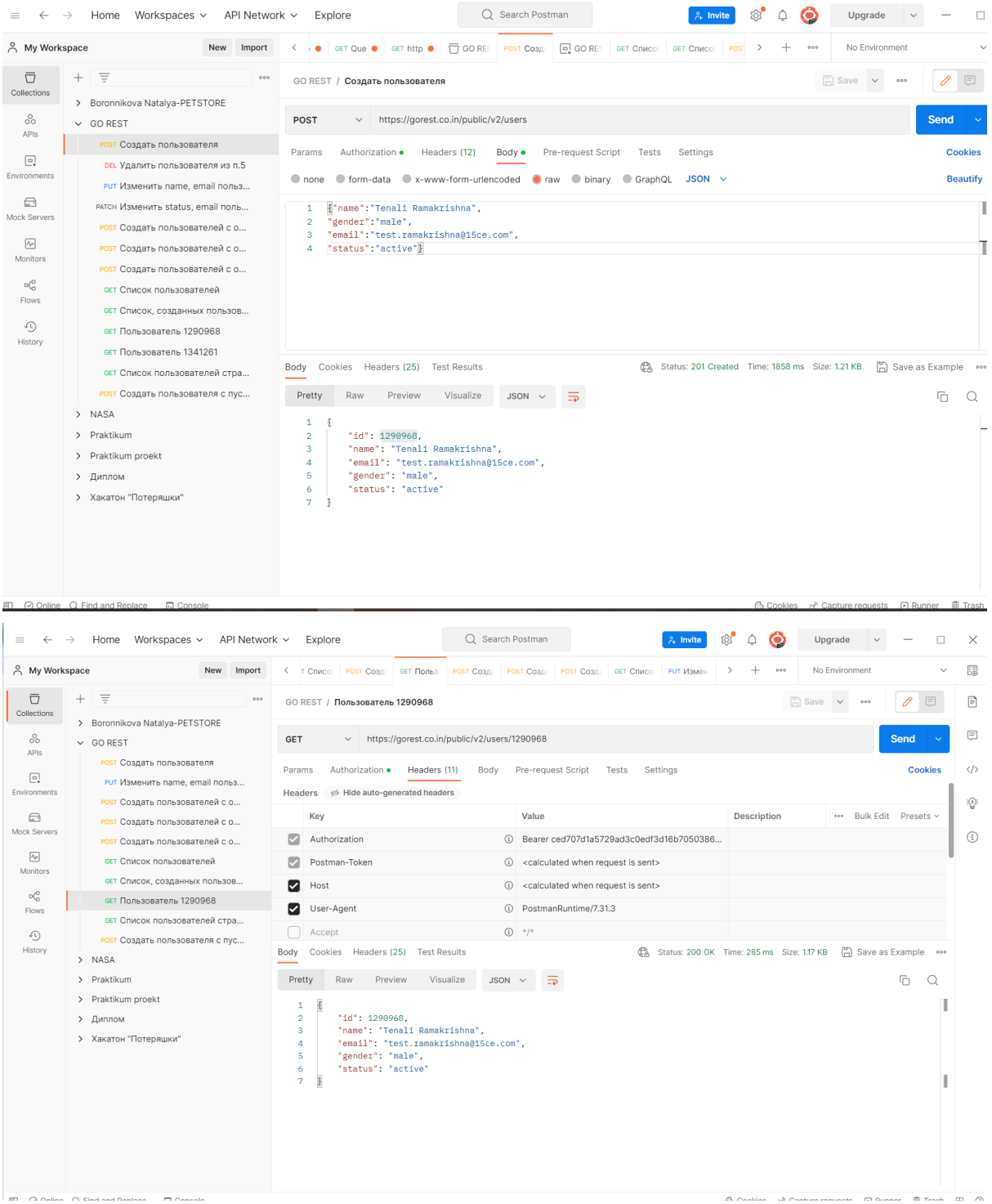
2. Тело ответа состоит из массива объектов, в объекте содержится ин-я об id, name, email, status, gender.



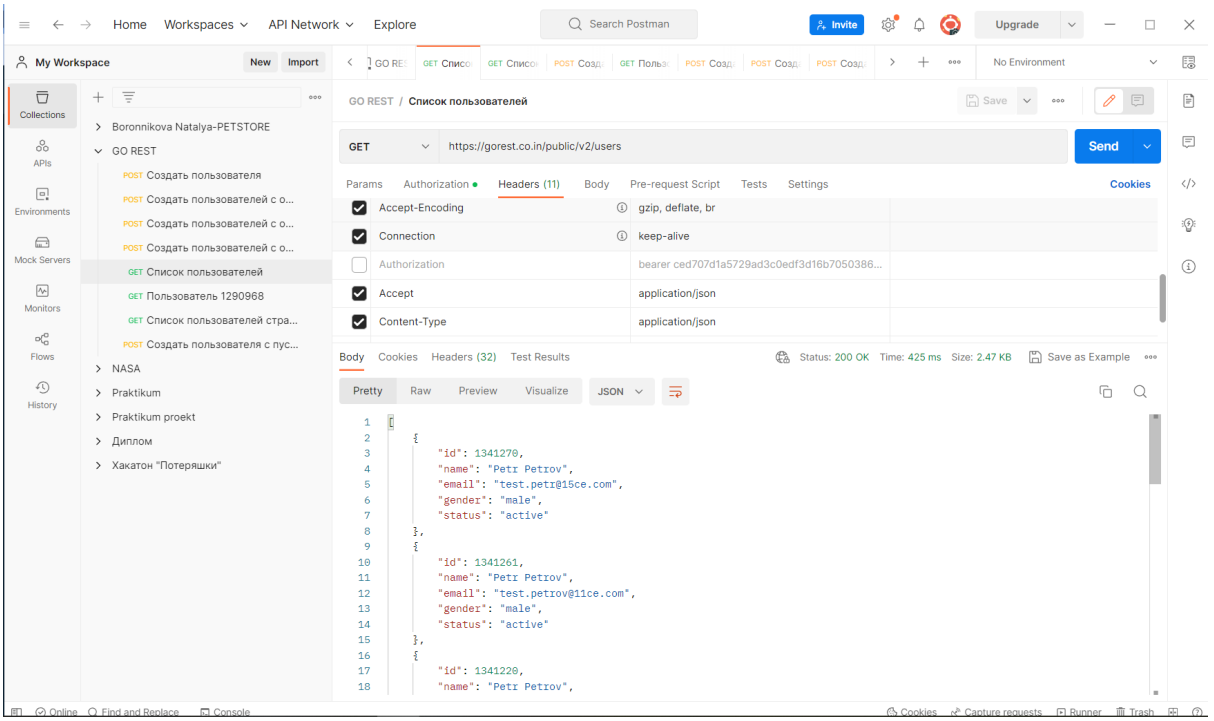
3. Во втором запросе в url указан параметр - страница 12.
4. При отправке запроса POST без тела возвращается ошибка 422, т.к. тело с данными пользователя не указано, сервер не может обработать запрос на создание пользователя. В теле ответа сообщение о том, что обязательные параметры не должны быть пустыми.



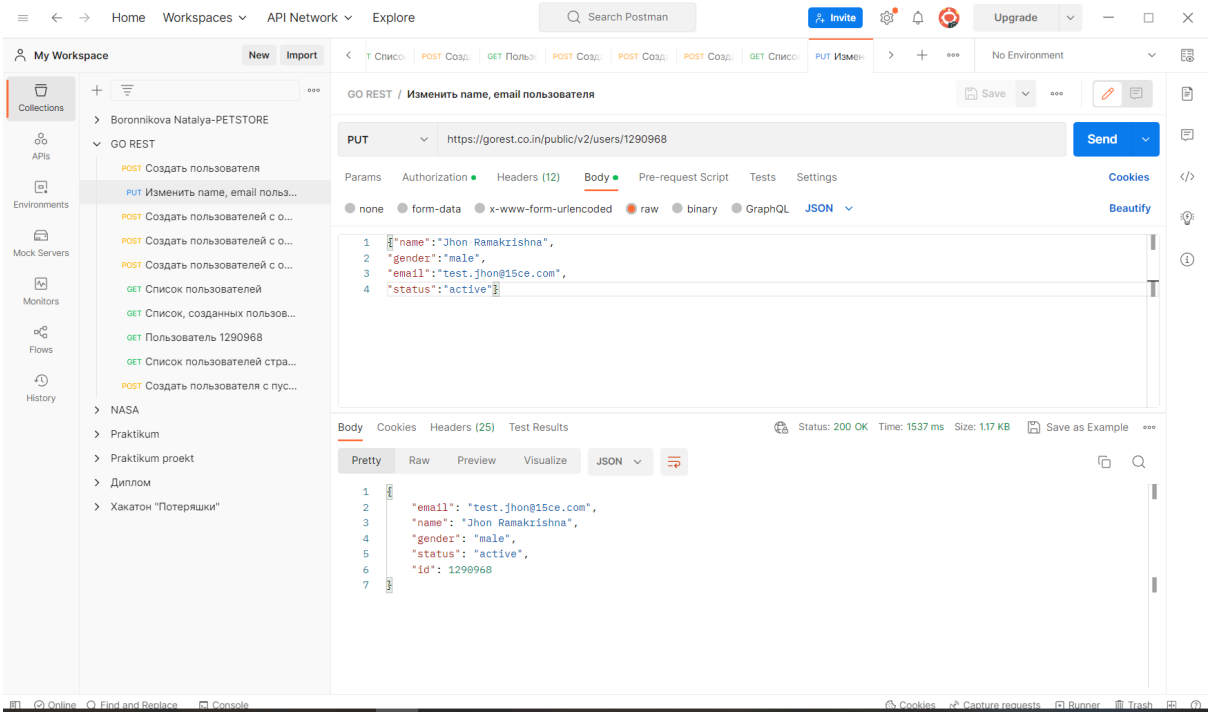
5. Статус ответа 201 Created. В теле ответа содержится id, name, email, status, gender.

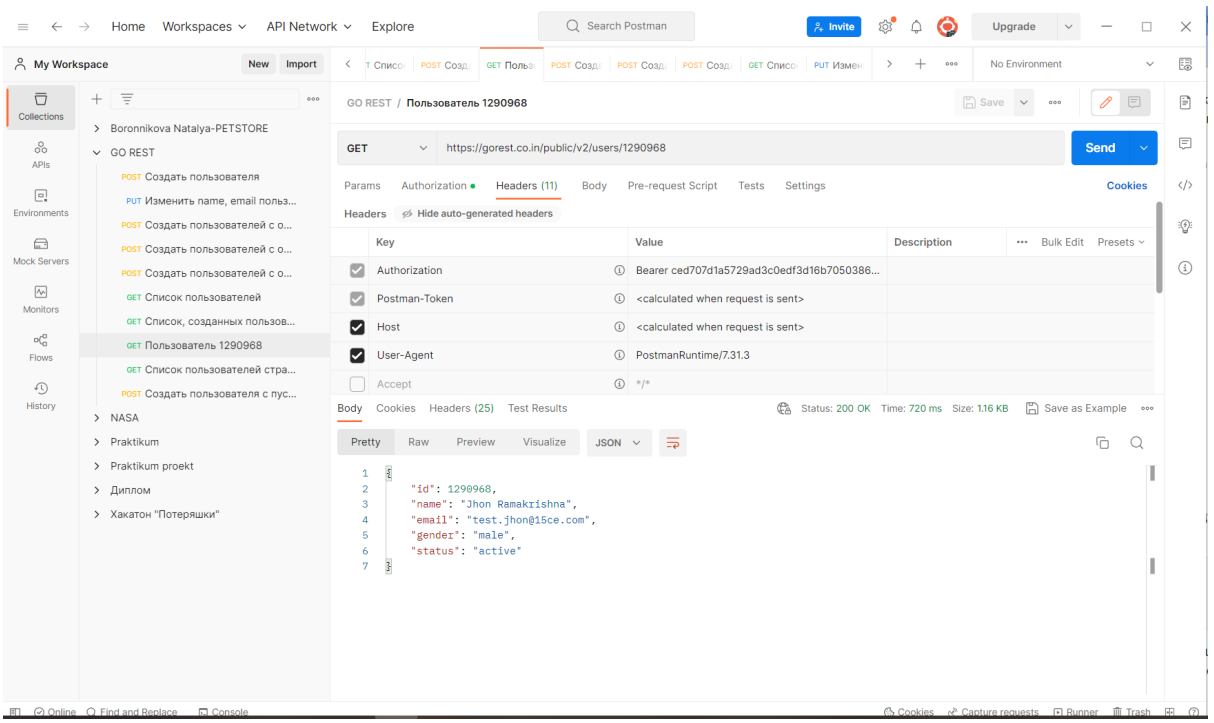
6. 

7. Можно получить пользователей списком как на скрине. Можно получить каждого пользователя отдельно, указав в url id пользователя (через параметры ключ - id, значение - номер)

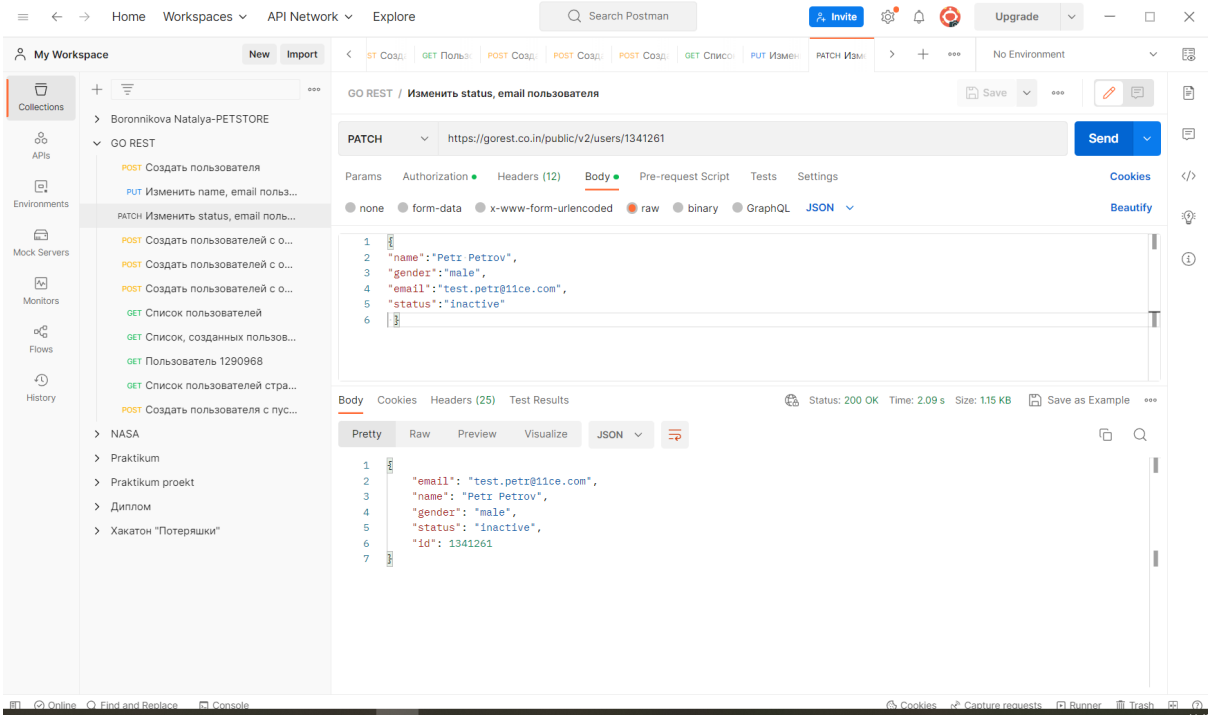

Home
Workspaces
API Network
Explore
Search Postman
Invite
Upgrade
My Workspace
New
Import
No Environment
Collections
APIs
Environments
Mock Servers
Monitors
Flows
History
Boronnikova Natalya-PETSTORE
GO REST
POST Создать пользователя
GET Список пользователей
GET Пользователь 1290968
NASA
Praktikum
Praktikum proekt
Диплом
Хакатон "Потеряшки"
GO REST / Список пользователей
Save
GET
https://gorest.co.in/public/v2/users
Send
Params
Authorization
Headers (11)
Body
Pre-request Script
Tests
Settings
Cookies
Accept-Encoding
gzip, deflate, br
Connection
keep-alive
Authorization
bearer ced707d1a5729ad3c0edf3d16b7050386...
Accept
application/json
Content-Type
application/json
Body
Cookies
Headers (32)
Test Results
Status: 200 OK
Time: 425 ms
Size: 2.47 KB
Save as Example
Pretty
Raw
Preview
Visualize
JSON
{
"id": 1341270,
"name": "Petr Petrov",
"email": "test.petr@15ce.com",
"gender": "male",
"status": "active"
},
{
"id": 1341261,
"name": "Petr Petrov",
"email": "test.petrov@11ce.com",
"gender": "male",
"status": "active"
},
{
"id": 1341220,
"name": "Petr Petrov",

8.

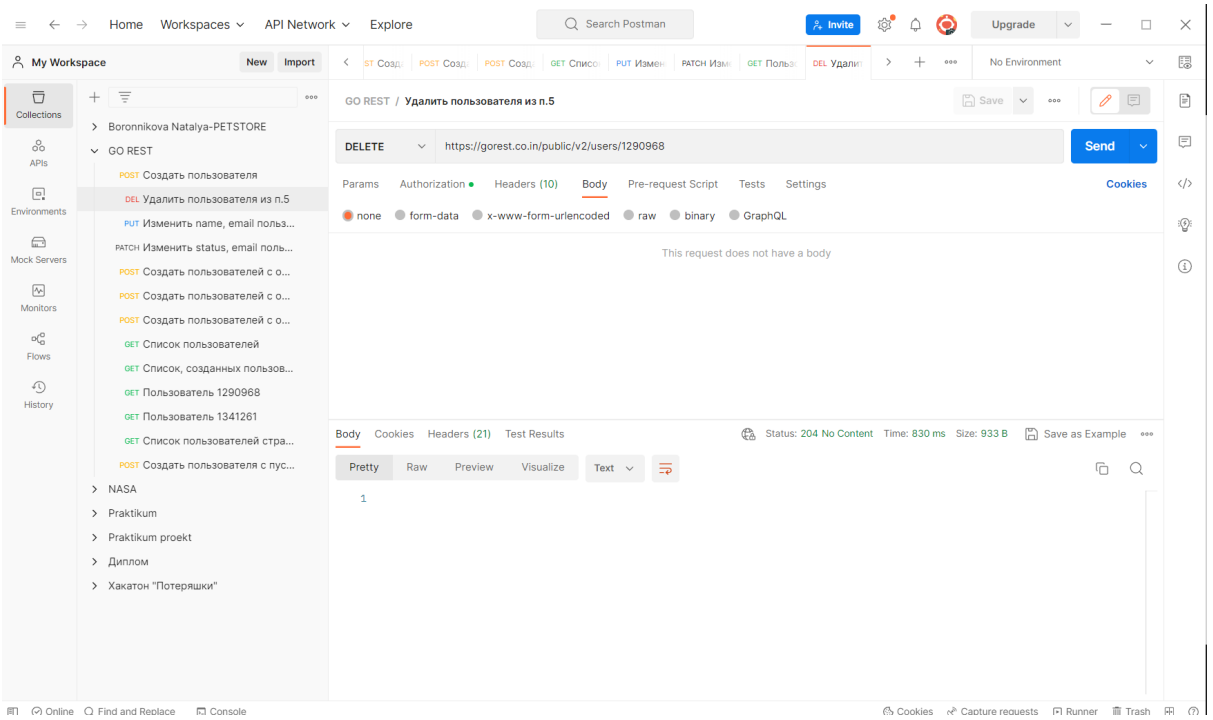

Home
Workspaces
API Network
Explore
Search Postman
Invite
Upgrade
My Workspace
New
Import
No Environment
Collections
APIs
Environments
Mock Servers
Monitors
Flows
History
Boronnikova Natalya-PETSTORE
GO REST
POST Создать пользователя
PUT Изменить name, email польз...
GET Список, созданных пользов...
GET Пользователь 1290968
NASA
Praktikum
Praktikum proekt
Диплом
Хакатон "Потеряшки"
GO REST / Изменить name, email пользователя
Save
PUT
https://gorest.co.in/public/v2/users/1290968
Send
Params
Authorization
Headers (12)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{"name":"Jhon Ramakrishna",
"gender":"male",
"email":"test.jhon@15ce.com",
"status":"active"}
Body
Cookies
Headers (25)
Test Results
Status: 200 OK
Time: 1537 ms
Size: 1.17 KB
Save as Example
Pretty
Raw
Preview
Visualize
JSON
{
"email": "test.jhon@15ce.com",
"name": "Jhon Ramakrishna",
"gender": "male",
"status": "active",
"id": 1290968
}



9. PUT полностью обновляет данные сущности. Если сущности нет, то создает новую как POST. В отличие от PATCH метод является идемпотентным. PATCH частично обновляет данные. Не является идемпотентным.



10. Тела ответа нет.



404 Not Found, в теле сообщения - ресурс не найден. Потому что пользователь удален, а метод GET идемпотентный.