**Первая часть задания выполняется в Postman.**

Загрузите решение в виде архива тестового файла с ответами на вопросы, скриншотами процесса и готовой коллекцией запросов Postman и пришлите ссылку на него.

**Подготовительный этап:**

Авторизуйтесь на gorest.co.in и получите токен для аутентификации (http bearer tokens)

**Задание:**

1. Получите список пользователей.

Каким запросом получили список пользователей?

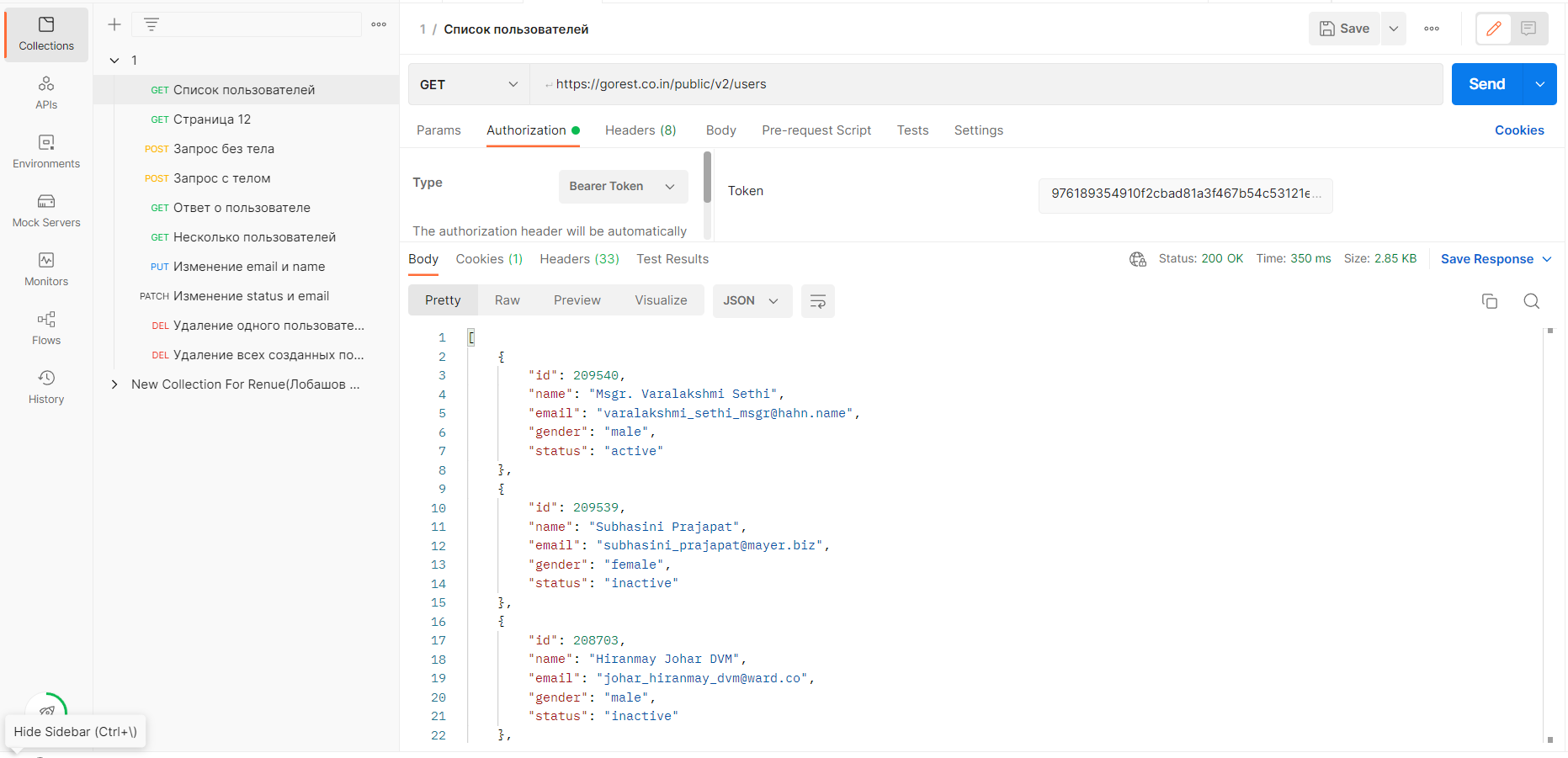
Какой статус ответа получили? Что содержится в теле ответа? Сделайте скриншот.

1.1

GET <https://gorest.co.in/public/v2/users>

200 ОК

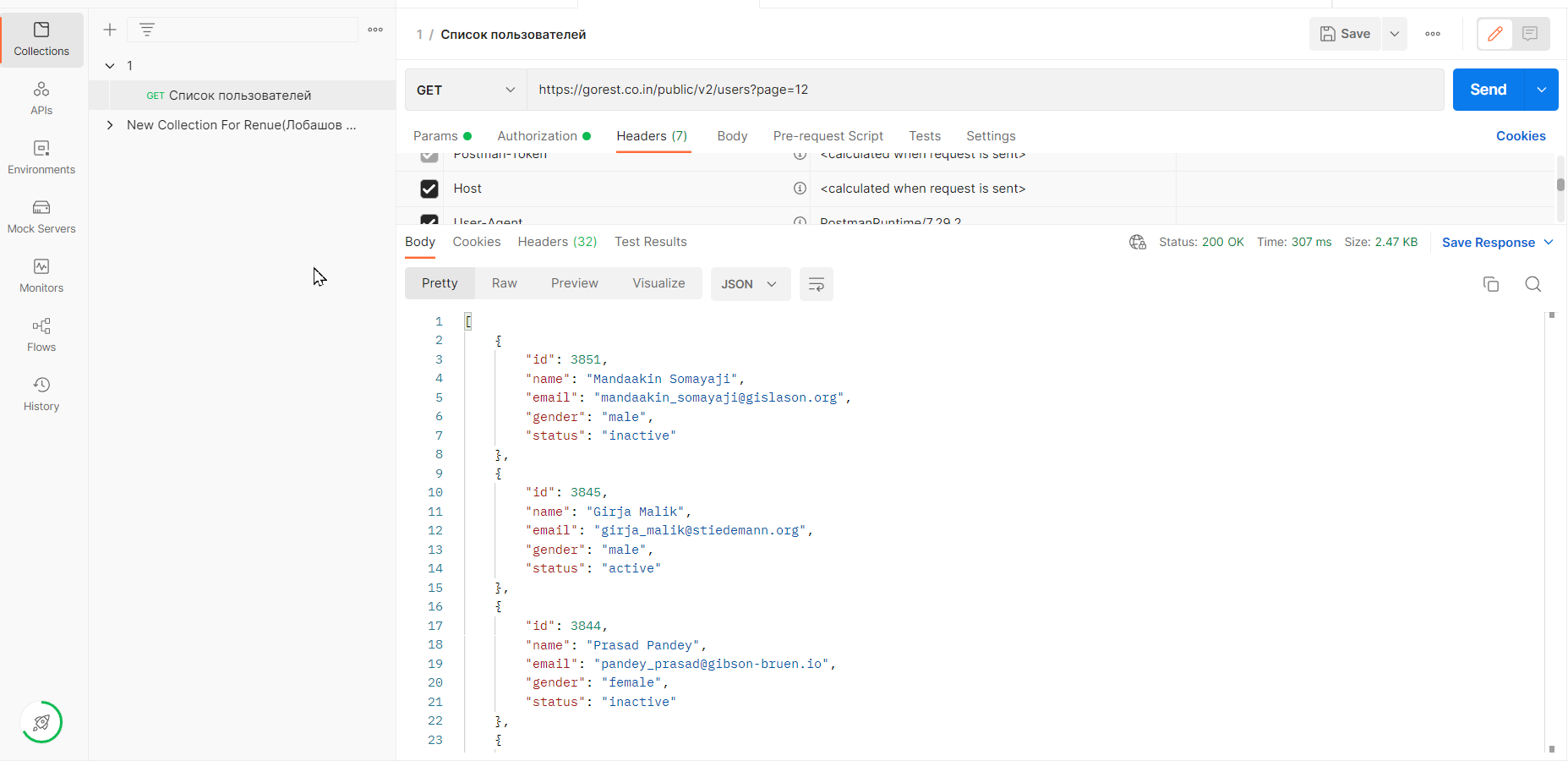
Id, имя, email, пол, статус по некоторым пользователям.



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

2.1

Пользователи со страницы 12



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

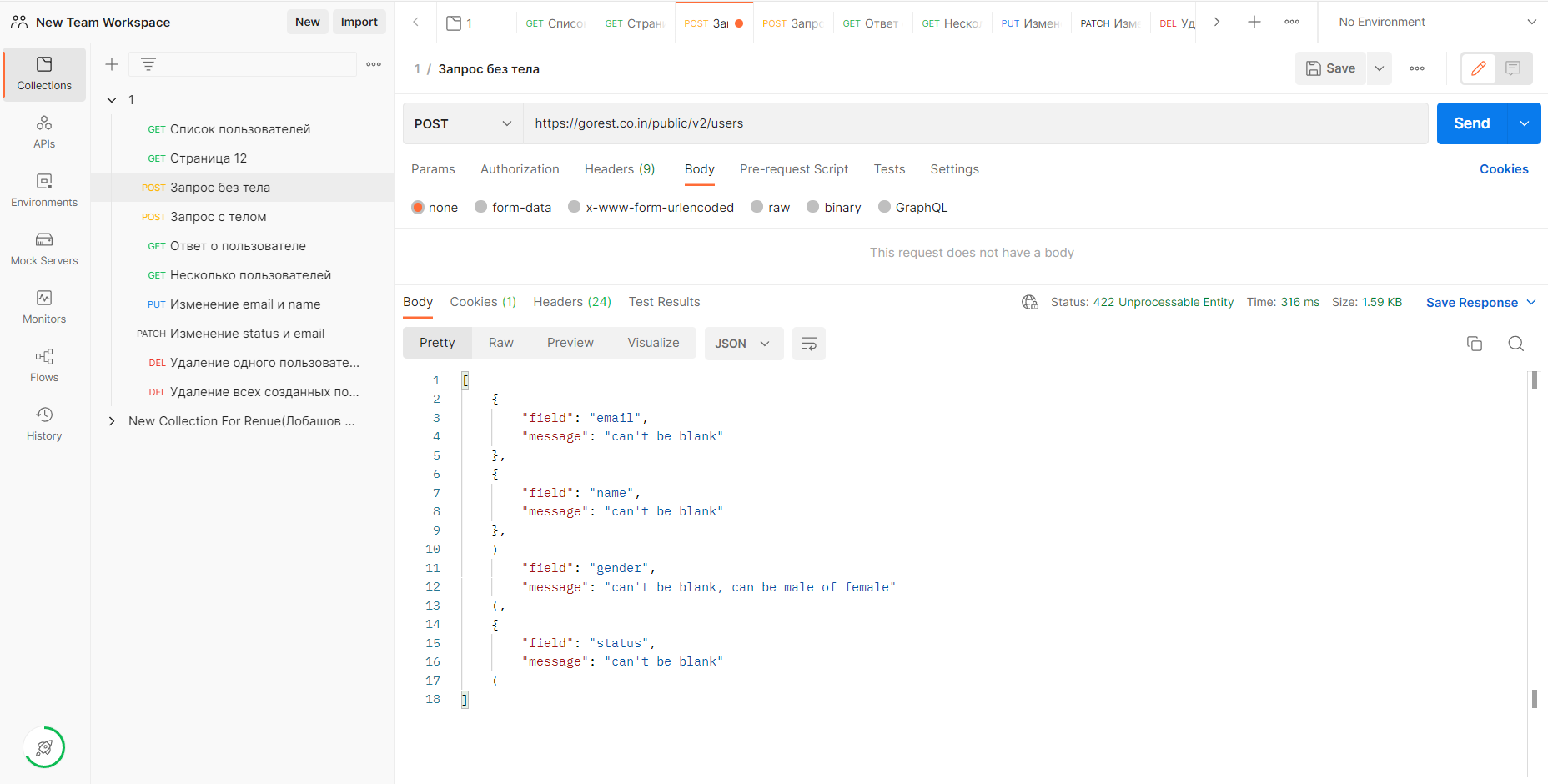
3.1

В п1. не указывается конкретная страница.

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

4.1

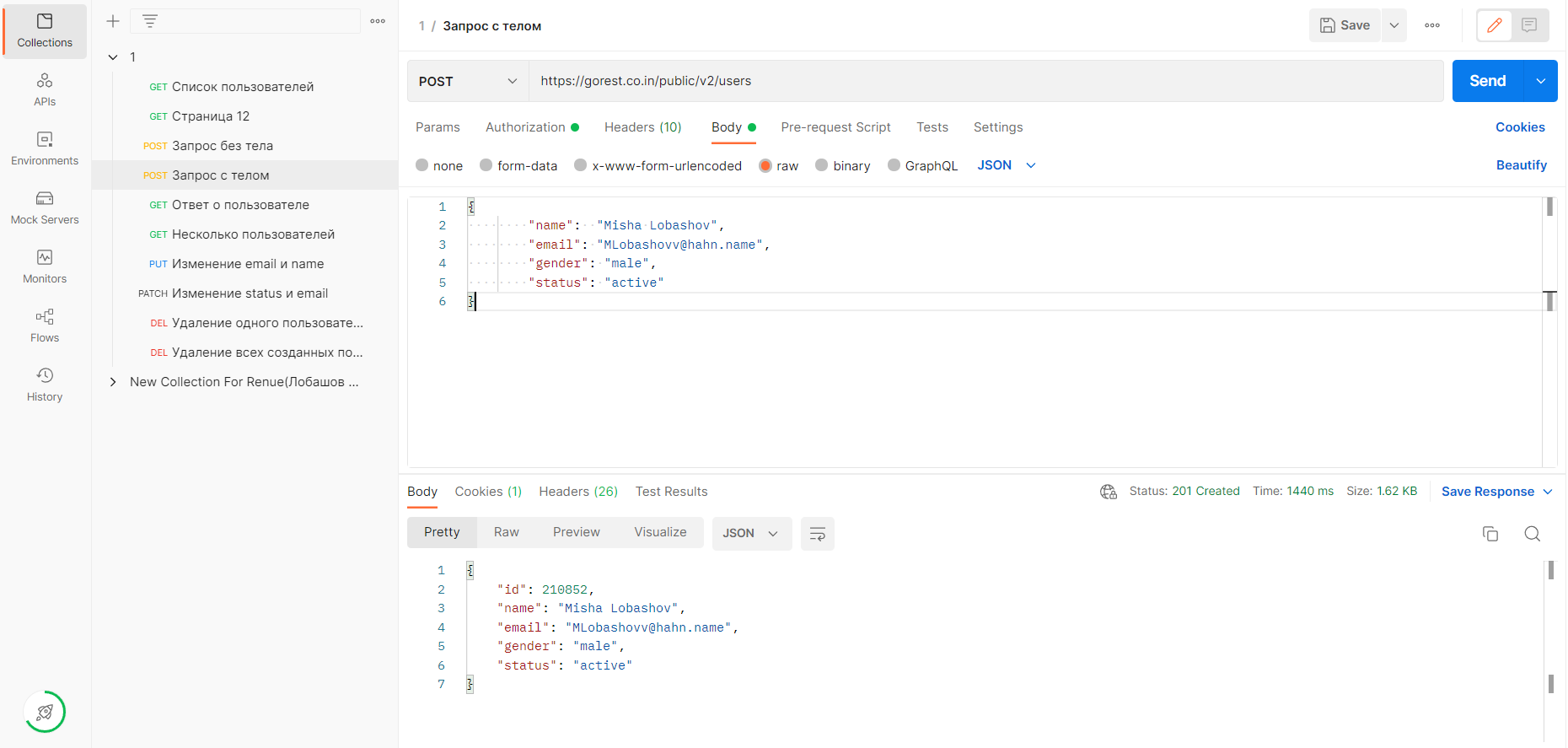
422 Unprocessable Entity - Сервер понимает запрос, но из-за того, что мы не указали имя, email, пол, статус - в ответе возвращает, то что поле не может быть пустым “can't be blank”



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

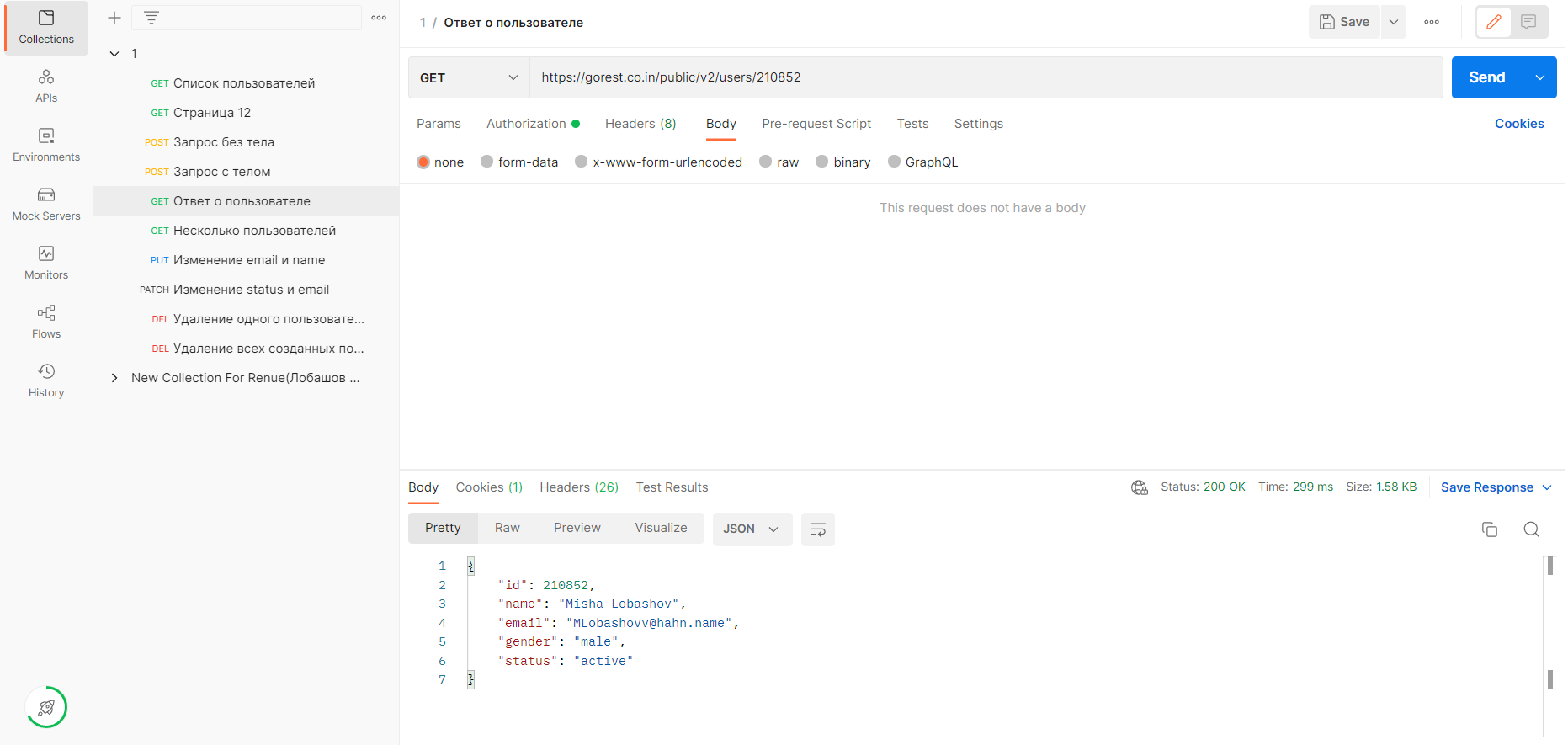
5.1

201 Created



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

6.1

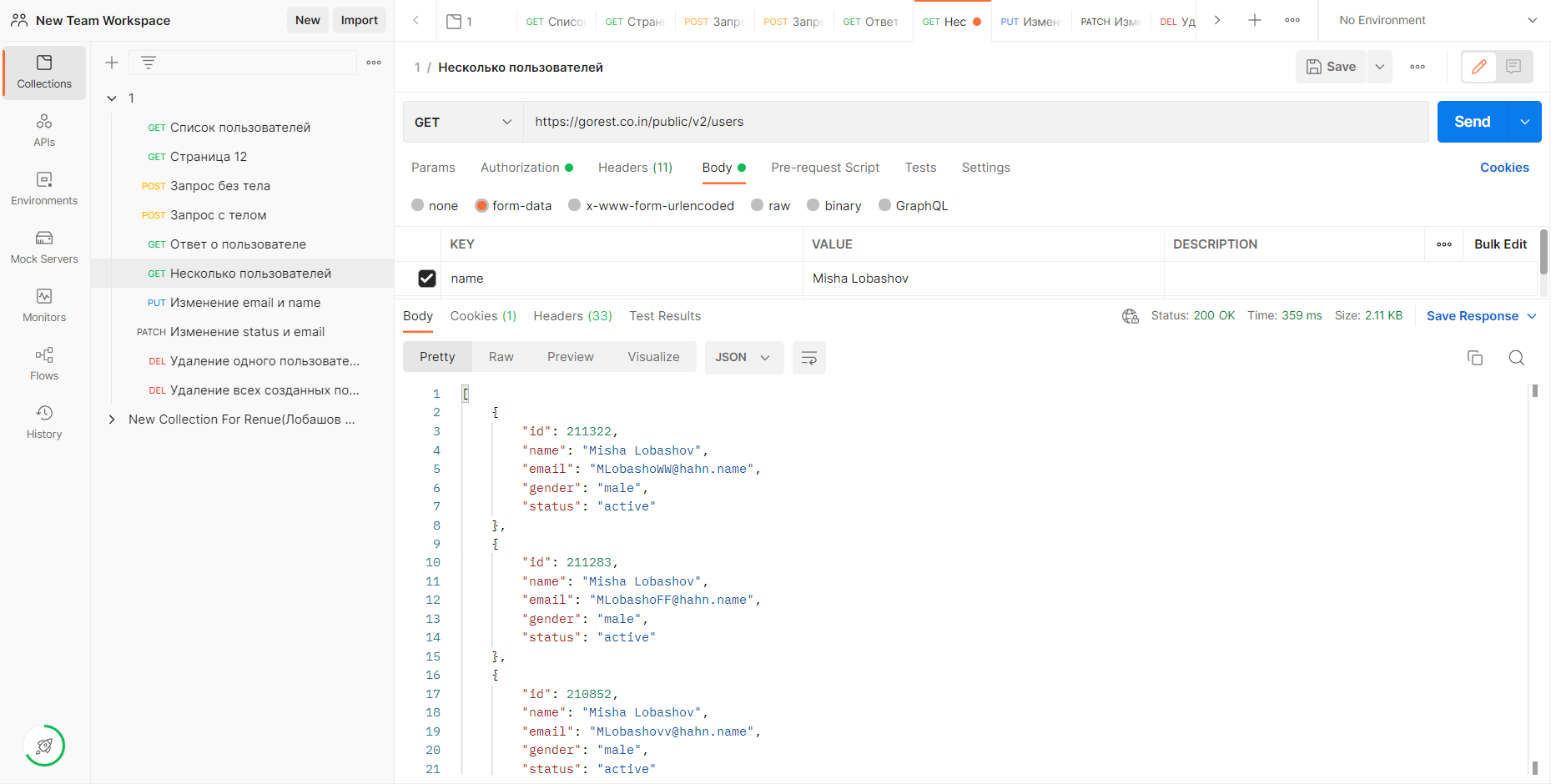


7. Создайте еще несколько пользователей с одинаковыми значениями “name”, но разными email. Получите список всех только что созданных пользователей. Сделайте скриншот.

Какими ещё get запросами можно получить запись о пользователе?

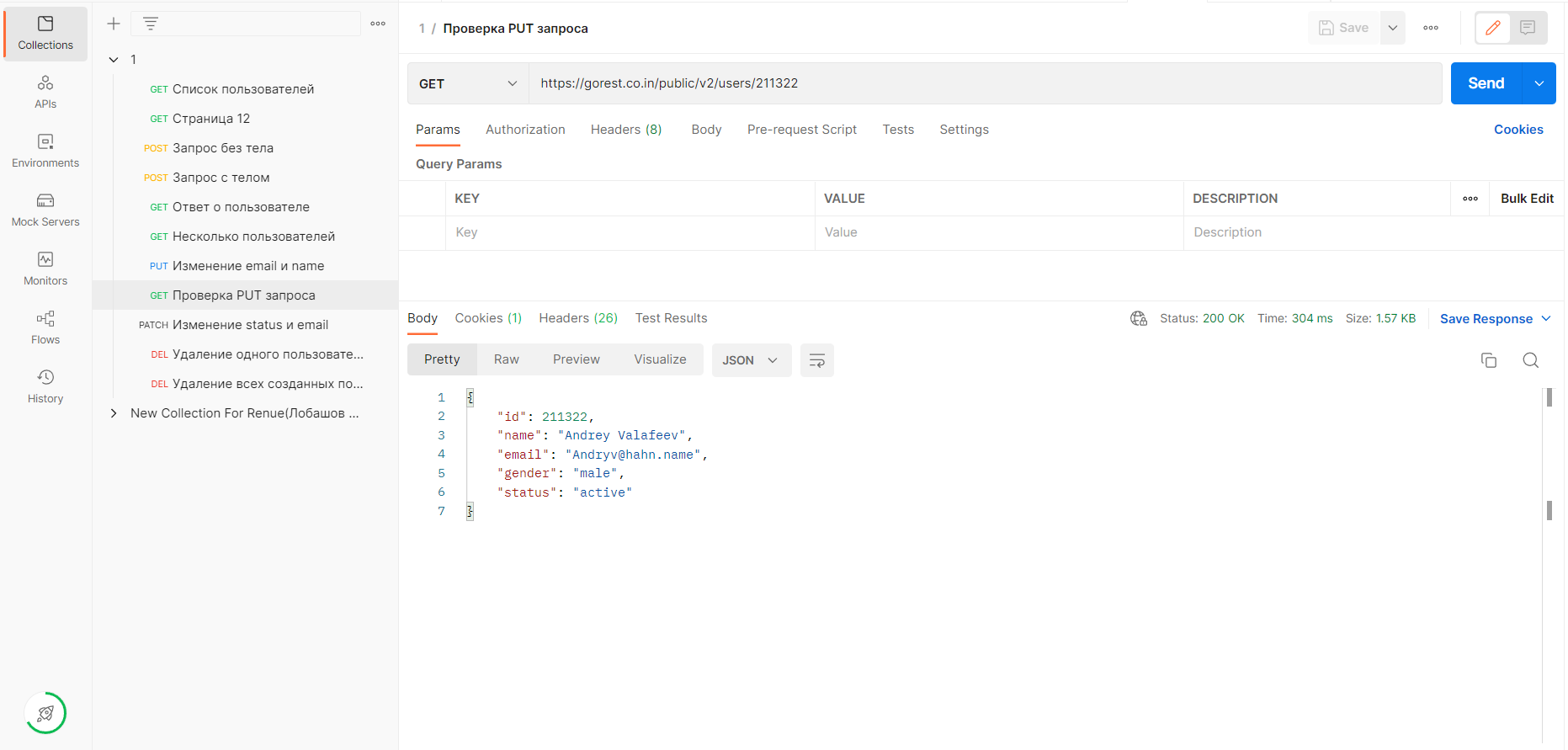
7.1

Найти через номер страницы, через номер id,



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

8.1



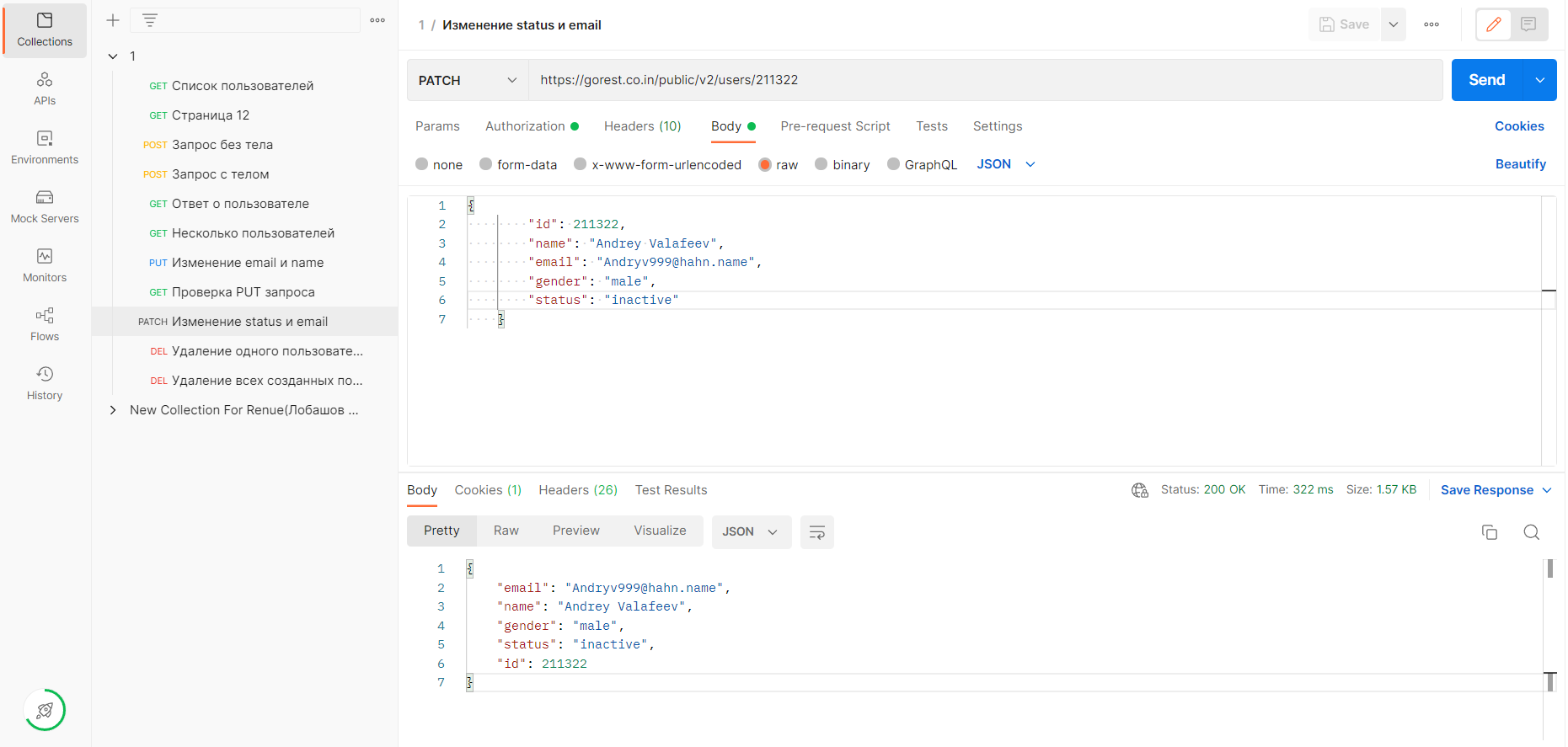
9. Чем отличаются запросы put и patch?

Напишите запрос с методом patch для корректировки полей status и email. Сделайте скриншот.

В чем отличие полученного результата, от результата запроса из п.8?

9.1

PUT - создает новый ресурс из старого, PATCH - частично меняет ресурс. Результат тот же, но метод другой.



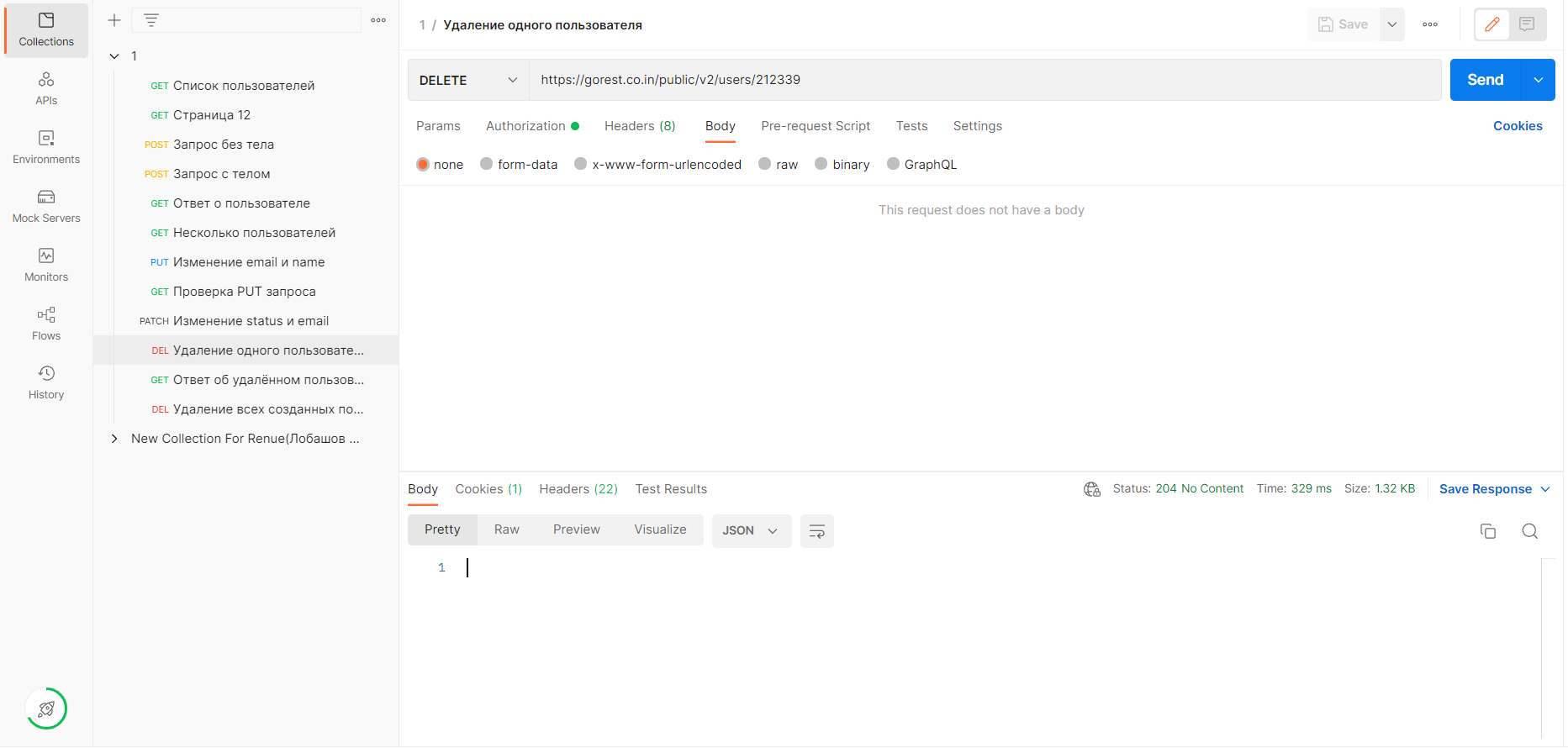
10. Удалите пользователя, созданного в п.5. Что содержится в теле ответа? Сделайте скриншот.

Выполните get запрос удаленного пользователя. Какой ответ пришел и почему? Сделайте скриншот.

10.1

204 No Content

404 Not Found - ресурс не найден, потому что мы его удалили.



11. Удалите всех ранее созданных пользователей.

12. Сохраните коллекцию получившихся запросов в Postman.

Кратко ответьте на вопросы ниже:

1. Опишите общий процесс тестирования отдельного модуля SPA приложения.

**Ответ:** Вопрос обширный и кратко на него не ответить. Если я мануальный тестировщик, то начал бы со сценариев тестирования, прошёлся бы по модулю позитивными и негативными тестами, заглянул в devtools, воспользовался инструментами для тестера, если нашёл ошибки - завел на них баг-репорты и составил документацию о проверках на модуле.

1. Что должен включать репорт найденной ошибки, какую структуру должен иметь?

**Ответ:** Номер, окружение, на котором была найдена ошибка, краткое и максимально описывающее ошибку название, шаги до бага, фактический и ожидаемый результат, серьезность, приоритет, скриншот/видео на котором будет видно ошибку, заметки (если нужно), автор баг-репорта и кому он направляется.

1. Как будет выглядеть баг-репорт, если, к примеру, не работает электрический чайник?

**Ответ:**

№342

Окружение:

г. Москва, ул Челнокова, д.34, оф 5 этаж 2, при входе налево.

Название:  
 При включении чайник шипит и не нагревает воду.  
Шаги:

1. Подключаю подставку чайника к розетке.
2. Набираю в чайник воды по отметку 1л.
3. Ставлю чайник на подставку и нажимаю “вкл”

Ожидаемый результат:

Чайник начинает кипятить воду

Фактический результат:

Чайник не нагревает воду, а шипит, на любые манипуляции не реагирует.

Серьезность:

Незначительная - от этого не зависит работа отдела.

Приоритет:

Низкий - есть дела намного важнее, (в бухгалтерии сломан компьютер, а сегодня нужно сдавать отчётность).



Автор: Лобашов М.О. Запрос: в отдел по закупке новых чайников

1. Составить баг репорт страниц и разделов, включая корневой по иерархии сайта https://site.domain.ru

**Ответ:**

№3

Окружение:

OC Windows 10 x 64; Браузер Opera GX; Версия: 91.0.4516.106 Stable

Тестируемый ресурс: [Ссылка](https://site.domain.ru)

Название:  
 Переходя по ссылке не удается получить доступ к сайту.  
Шаги:

1. Перехожу по ссылке
2. Чищу кэш
3. Использую DNS поверх HTTPS вместо системных настроек DNS

Ожидаемый результат:

Сайт открывается.

Фактический результат:

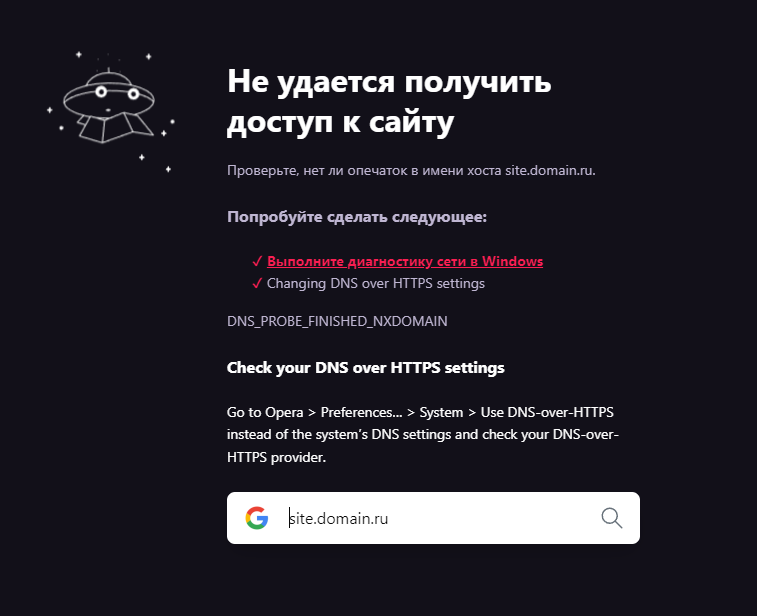
Сайт не открывается, на любые попытки исправить это, реагирует ошибкой не на стороне пользователя.

Серьезность:

Блокирующая - сайт не получается открыть

Приоритет:

Высокий - у пользователя не получится зайти на сайт



Автор: Лобашов М.О. Запрос: Алатушев В.И.

1. Необходимо протестировать регистрацию нового пользователя в приложении. Расскажите, как вы будете это делать? (описать все этапы)

**Ответ:**

* Пробую зарегистрировать нового пользователя, посмотрю появляется он в базе или нет, правильно ли там отобразится.
* Проверяю, получается ли войти в приложение.
* Пробую зарегистрировать этого же пользователя.
* Ввожу в поля невалидные данные
* Тестирую формы, кнопки и календари
* Вставляю во все формы - непотребства из буфера обмена (Тексты, картинки, полный номер телефона в форму +7, ссылки и тд.)
* Создать вторую учетную запись и ввести email от одной учётки, а пароль от другой.
* Кроссбраузерное тестирование
* UX
* Проверю, есть ли отправка сообщения на email об успешной регистрации
* Создам репорты по всем багам, отправлю на исправление.
* Сделаю отчёт о проведённой работе.

1. Представьте ситуацию, что у разрабатываемого приложения еще нет интерфейса, но реализован REST API. Разработчик просит вас создать какую-то сущность в базе и проверить, что она создалась с нужными параметрами. Опишите ход ваших действий в данной ситуации: что и как вы бы проверили, опираясь на имеющееся описание API, с указанием конкретики (название типов запросов и т.д.)

**Ответ:**

Если я правильно понимаю вопрос, посмотрел бы описание по API и по описанию создавал бы запросы и проверял ответы на них.

1. Поле username должно быть обязательным, но оно не обязательно. Приведите пример баг-репорта, созданного для этой ошибки.

**Ответ:**

№850

Окружение:

OC Windows 10 x 64; Браузер Opera GX; Версия: 91.0.4516.106 Stable

Название:  
 Поле username должно быть обязательным, но оно не обязательно..  
Шаги:

1. Перехожу по ссылке на форму авторизации.
2. Заполняю все формы, кроме username
3. Нажимаю на кнопку “регистрация”

Ожидаемый результат:

Появится уведомление с ошибкой, ещё до нажатия кнопки “регистрация”.

Фактический результат:

Получилось зарегистрироваться без username, никакого уведомления об ошибке не появилось, могу пользоваться сайтом.

Серьезность:

Значительная

Приоритет:

Средний

1. Можно ли передавать в json бинарные данные? Почему?

**Ответ:** Можно, если закодировать в текстовые строки.

1. Написать тестовые наборы данных для поля ввода даты, которое отсеивает пользователей в возрасте до 18 лет.

**Ответ:**

* Корректные значения
* Некорректные
* Отрицательные числа
* 1/0
* Не числовые значения
* Эквивалентное разбиение/ граничные значения
* Пустая форма
* Дробные числа

1. Как протестировать скачивание файла при недостаточном количестве места на диске.

**Ответ:**

Можно протестить скорость загрузки, скорость отдачи, разные форматы файлов, скачать файл не на диск.

1. Как протестировать загрузку файла при недостаточном количестве места на сервере.

**Ответ:**

Так же скорость. Возможно посмотреть, как будет себя вести загрузка, если обновить сервер

1. Есть веб-страница с полями: e-mail, password и кнопкой submit. Необходимо привести примеры отрицательных тест-кейсов, по которым можно проверить эту страницу.

**Ответ:** Если в вопросе говориться о негативных тест-кейсах, то:

* Не заполняем поля e-mail, password, или заполняем только 1 поле.
* Кнопка submit сама по себе не активна
* Выполняем проверку на максимальное количество символов в полях e-mail, password
* Заполняем невалидные данные, производится вход
* Отсутствие @ в email
* Email с пробелами в имени пользователя
* Email без доменной части
* Email с кириллическим доменным именем (mail@всемпривет.ру)

И тд.

1. Имеется функционал калькулятора, доступный через веб браузер по ссылке. Он имеет только функцию делить, так сказать, MVP-версию. Диапазоны для вписывания в числитель и делитель от 0,1 до 99,9. Вывод значения происходит автоматически, потому что front-end реализован на React JS. Как вы будете тестировать этот функционал? Какие виды тестирования примените? Какие техники тест-дизайна используете?

**Ответ:**

Вручную пройдусь по функции деления, опять же целые, отрицательные, не числовые и тд, применю ручное тестирование, писать автотесты по одной функции калькулятора как-то странно. Применю эквивалентное разбиение, причину и следствие, предугадывание ошибок.