

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

**Дисциплина:** Бэк-энд разработка

Отчет

Домашняя работа №4

Выполнил:

Шалунов Андрей

Группа К3340

Проверил:

Добряков Д. И.

Санкт-Петербург

2025 г.

## Задача

Реализовать тестирование API средствами Postman.

Написать тесты внутри Postman.

## Ход работы

Создание окружения

В Postman настроено окружение RealEstate с переменными:

baseUrl = http://localhost:8000/api

jwtToken — (будет заполняться после логина)

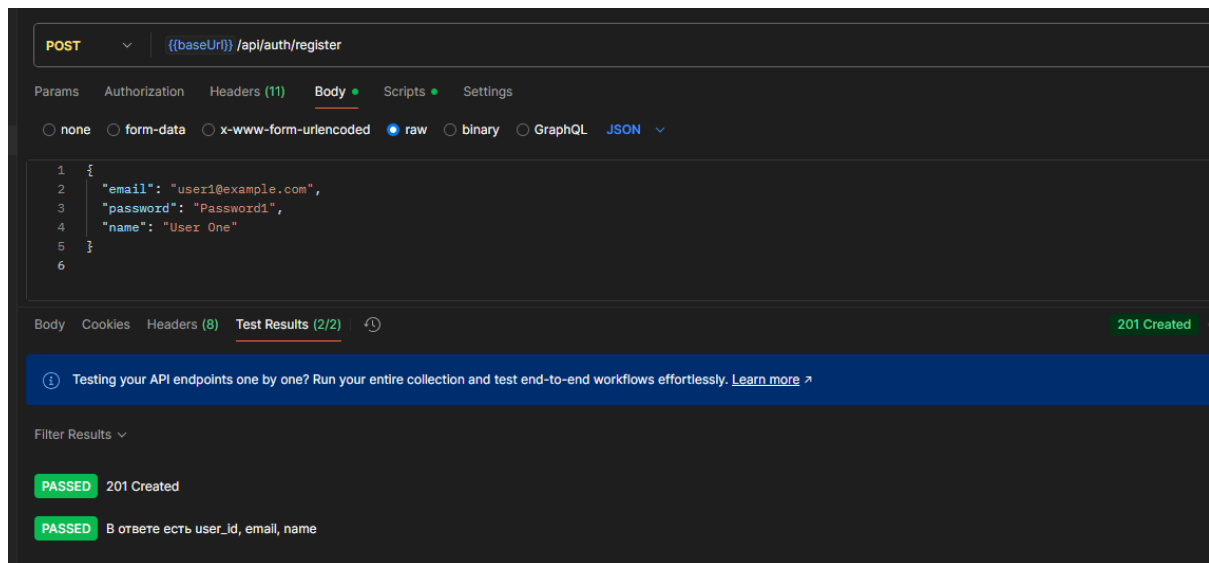
Тестовые сценарии

### 1. Регистрация пользователя

Метод: POST {{baseUrl}}/auth/register

Tests:

```
pm.test("201 Created", () => {  
    pm.response.to.have.status(201);  
});  
  
const json = pm.response.json();  
  
pm.test("В ответе есть user_id, email, name", () => {  
    pm.expect(json).to.have.all.keys("user_id", "email", "name");  
});
```



## 2. Логин и получение токена

Метод: POST {{baseUrl}}/auth/login

Tests:

```
pm.test("200 OK", () => {

  pm.response.to.have.status(200);

});

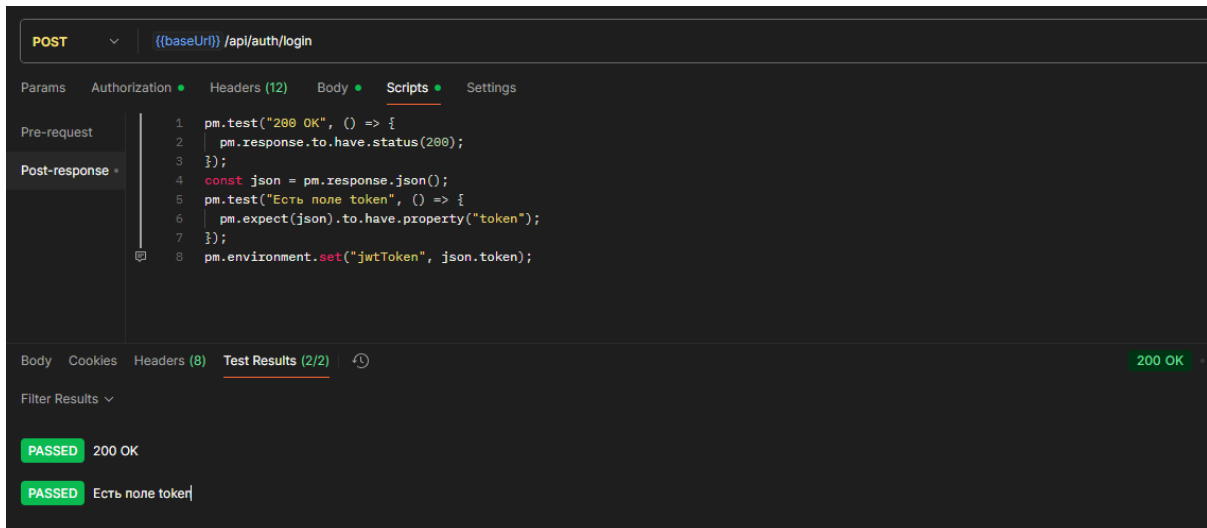
const json = pm.response.json();

pm.test("Есть поле token", () => {

  pm.expect(json).to.have.property("token");

});

pm.environment.set("jwtToken", json.token);
```

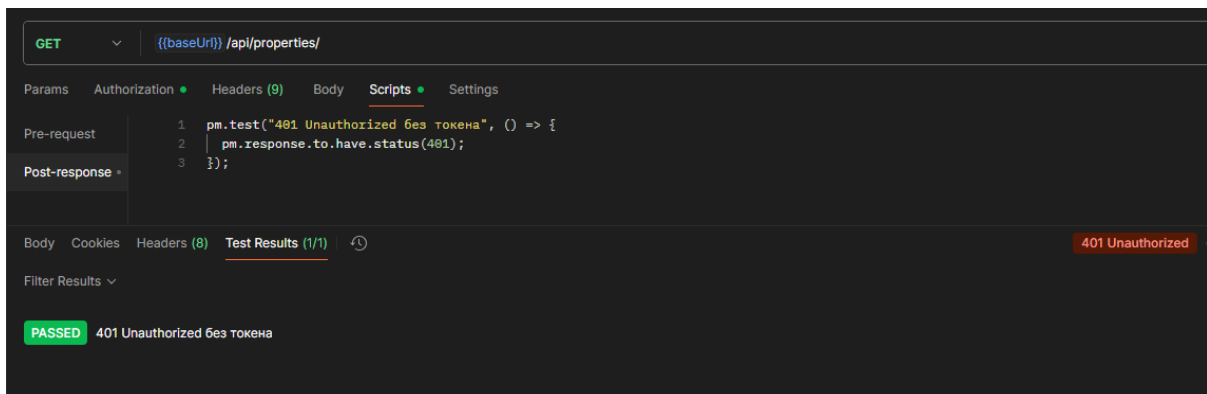


### 3. Получить все свойства (без авторизации)

Метод: GET {{baseUrl}}/properties

Tests:

```
pm.test("401 Unauthorized без токена", () => {
  pm.response.to.have.status(401);
});
```



### 4. Создать Booking

Метод: POST {{baseUrl}}/bookings

Auth: Bearer {{jwtToken}}

Tests:

```

pm.test("201 Created", () => {

  pm.response.to.have.status(201);

});

const b = pm.response.json();

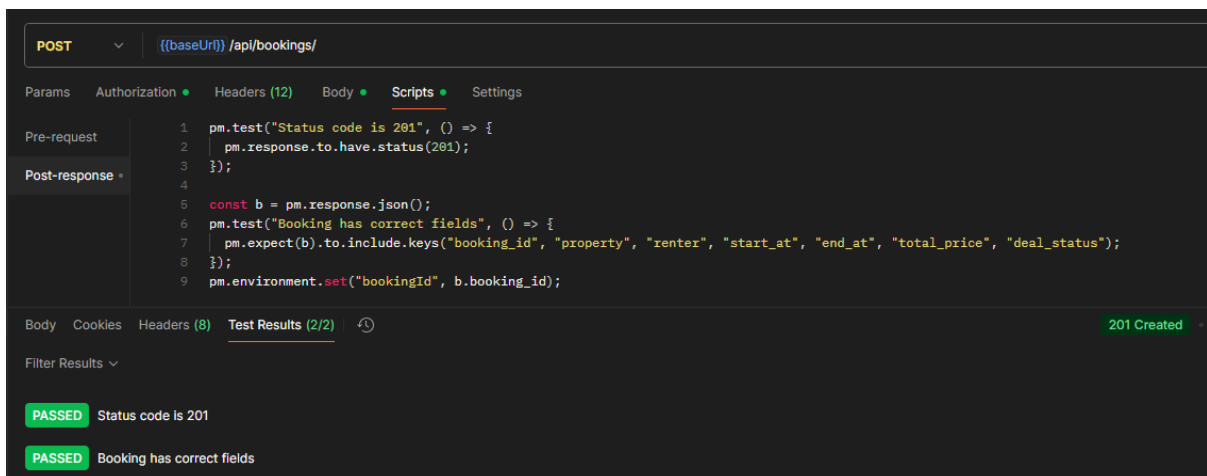
pm.test("Booking содержит корректные поля", () => {

  pm.expect(b).to.include.keys("booking_id", "start_at", "end_at", "total_price", "deal_status");

});

pm.environment.set("bookingId", b.booking_id);

```



## 5. Удаление Booking

Метод: DELETE `{{baseUrl}}/bookings/{{bookingId}}`

Auth: Bearer `{{jwtToken}}`

Tests:

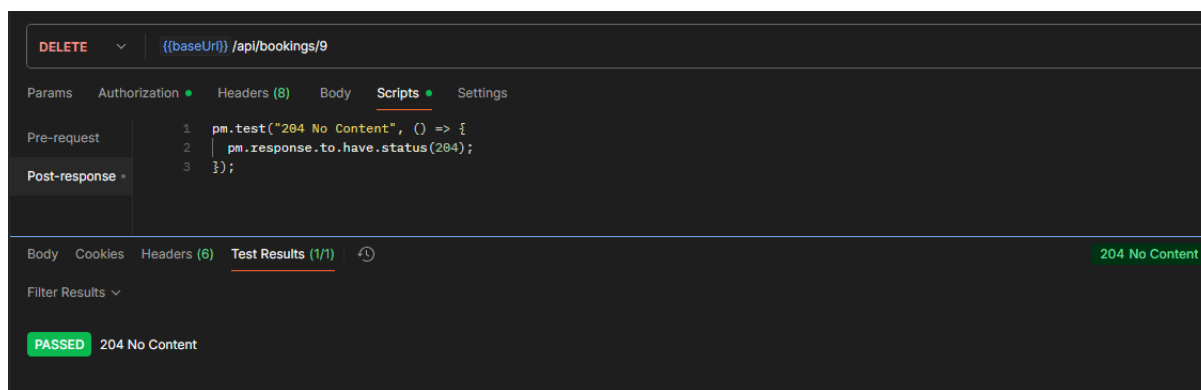
```

pm.test("204 No Content", () => {

  pm.response.to.have.status(204);

});

```



## Вывод

В ходе работы настроено окружение и создана коллекция из 5 ключевых тестовых сценариев для основных эндпоинтов API: регистрация, логин, проверка неавторизованного доступа, создание и удаление бронирования. Тесты проверяют коды ответов, структуру JSON и сохраняют в переменные необходимые данные (JWT-токен, ID бронирования) для последующих запросов.