**М****ИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ**

**Ордена Трудового Красного Знамени**

**Федеральное государственное бюджетное образовательное учреждение высшего образования**

**«Московский технический университет связи и информатики»**

Кафедра «Математическая кибернетика и информационные технологии»

**Отчет по практической работе**

по дисциплине «Введение в информационные технологии» на тему:

«Основы работы с FastAPI»

Выполнил: студент группы

БПИ2403

Батин Платон

Проверил:

Москва

2025

1. **Цель работы:**

Получить практические навыки в использовании FastAPI

1. **Ход работы:**

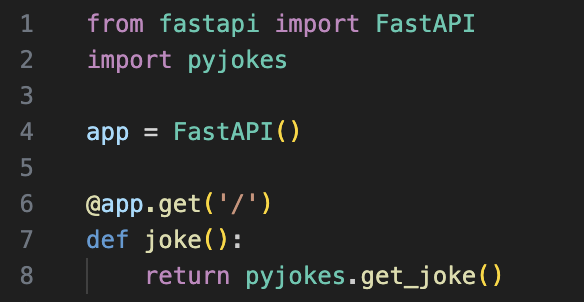
* Установка необходимых компонентов, а именно fastapi, uvicorn и pyjokes.

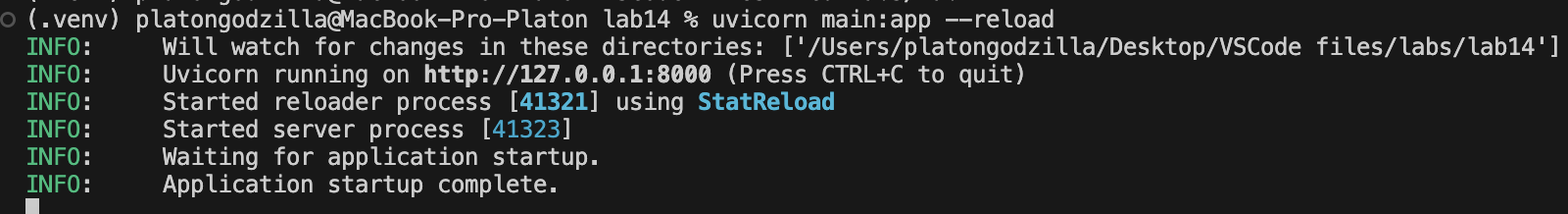
Команды для установки:

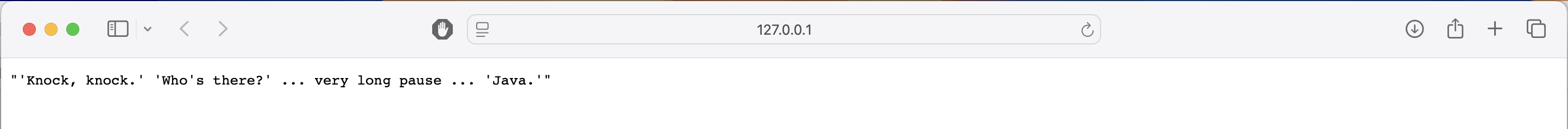
pip install fastapi

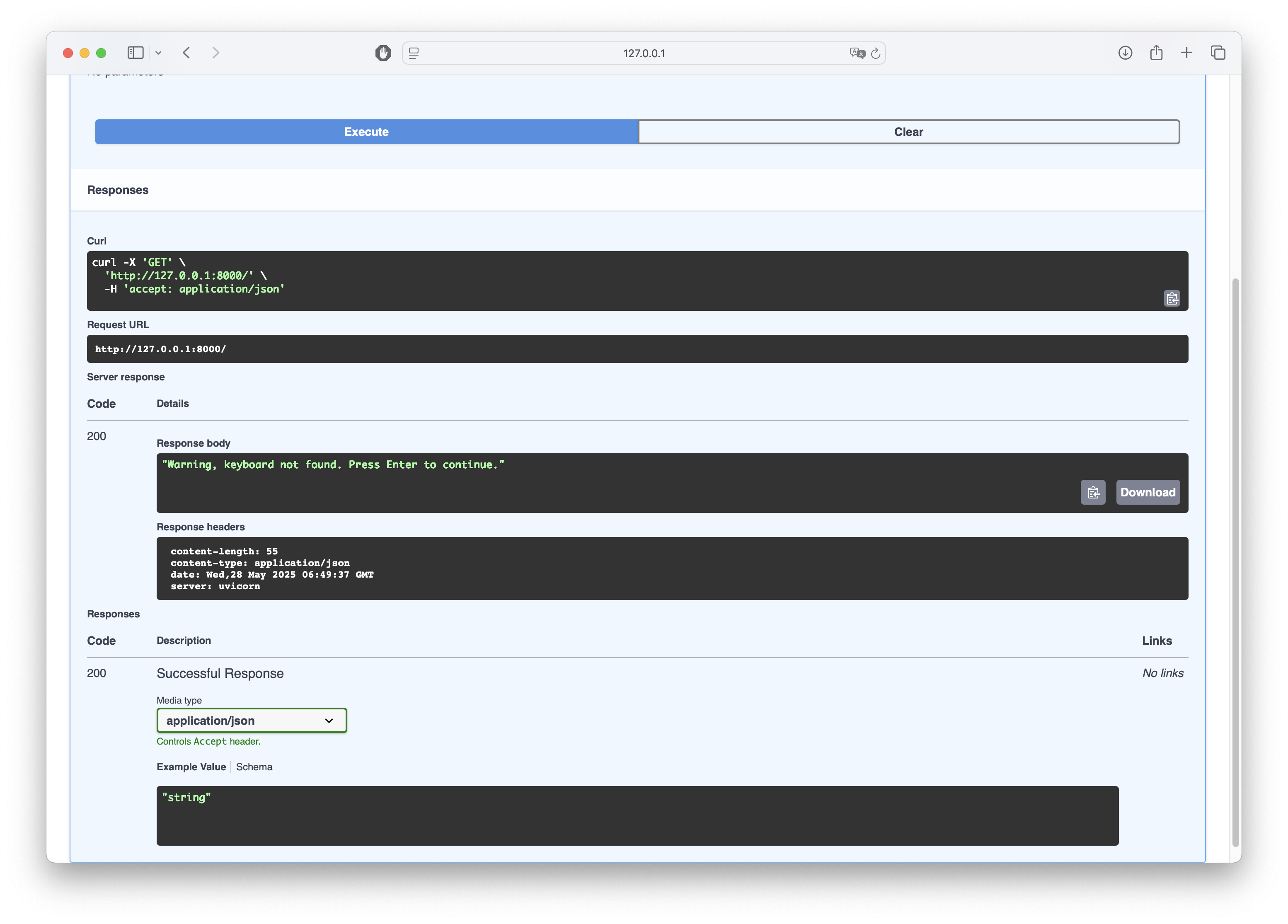
pip install uvicorn

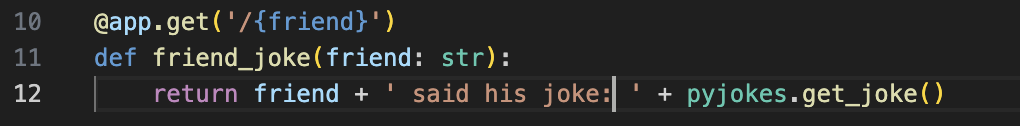
pip install pyjokes

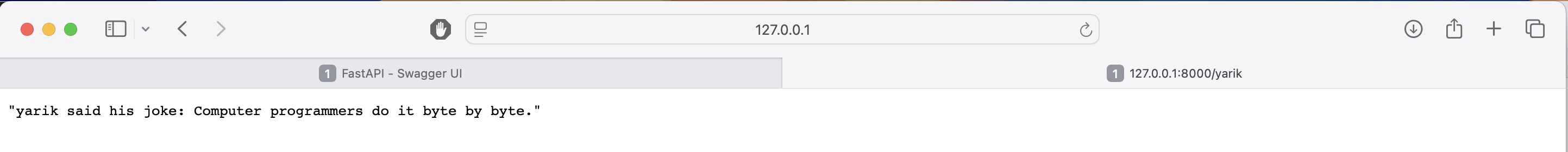
* Далее создаётся файл main.py для назначения роутов.

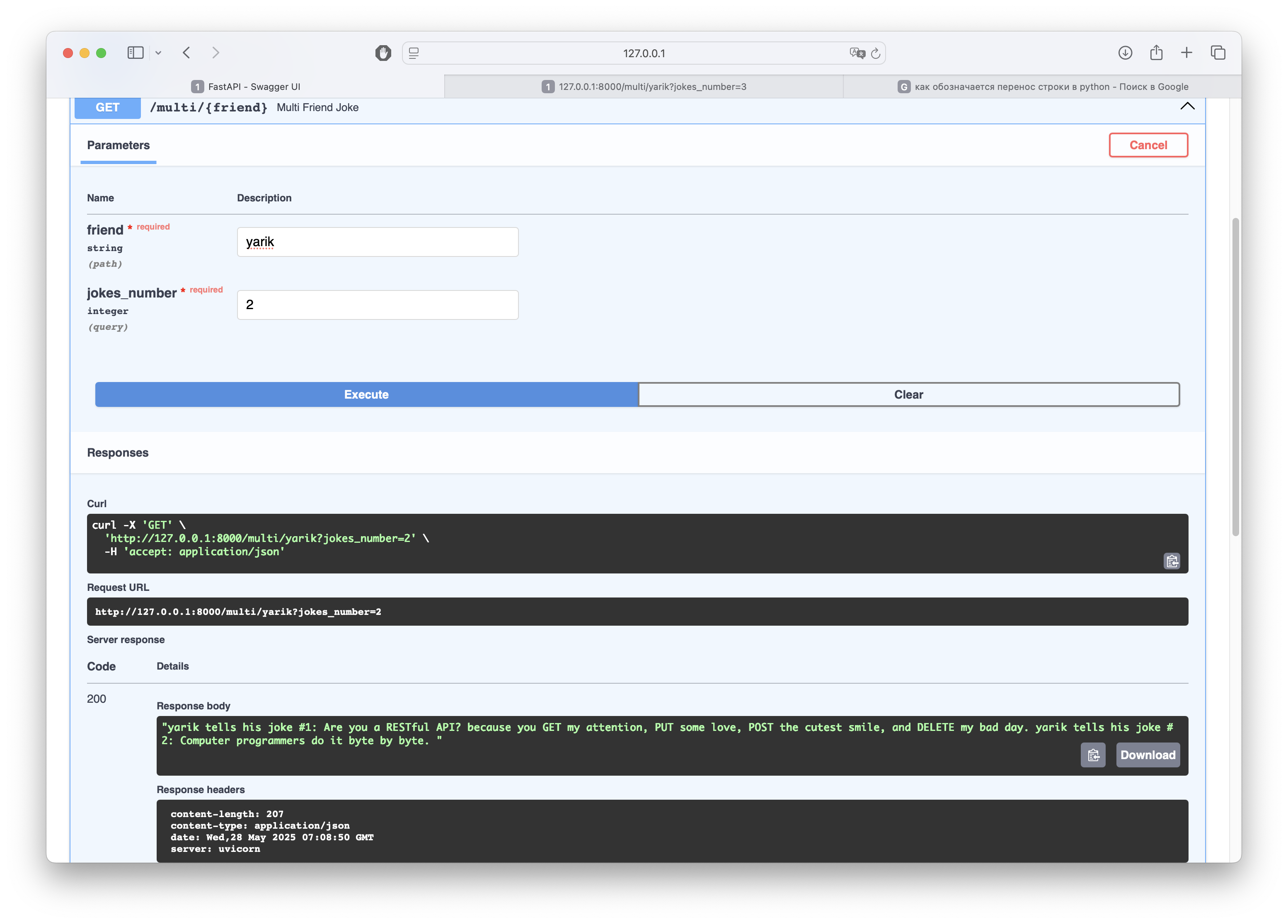
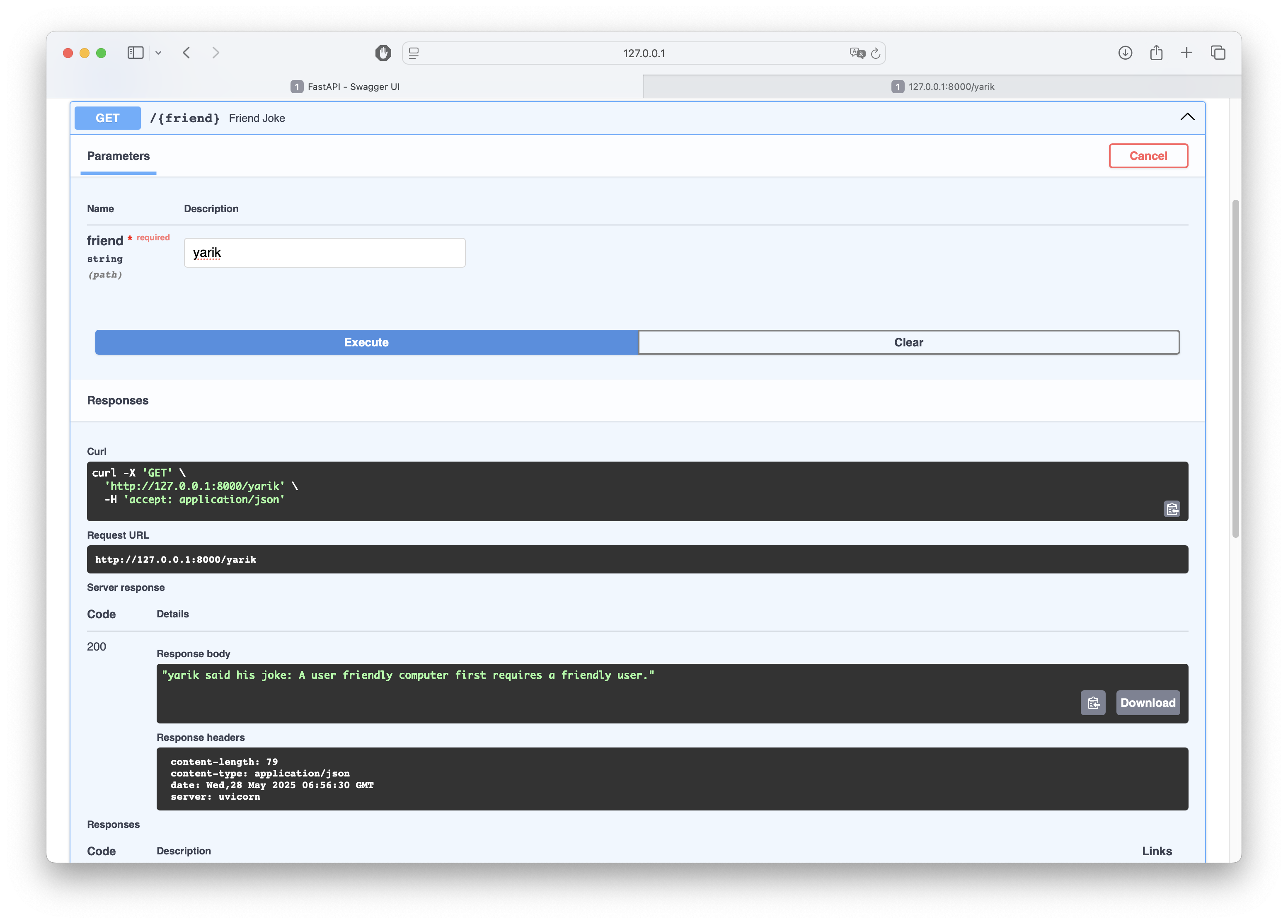
* После запускаем виртуальный сервер

Результат:

Получаем:

* Добавим роут для шутки друга

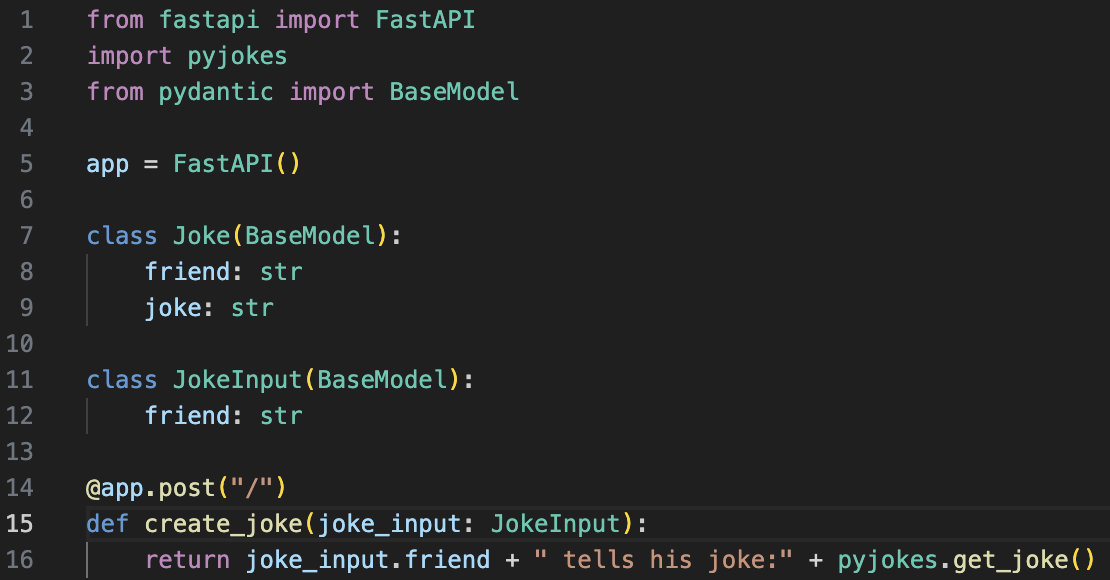
Например, имя друга будет yarik

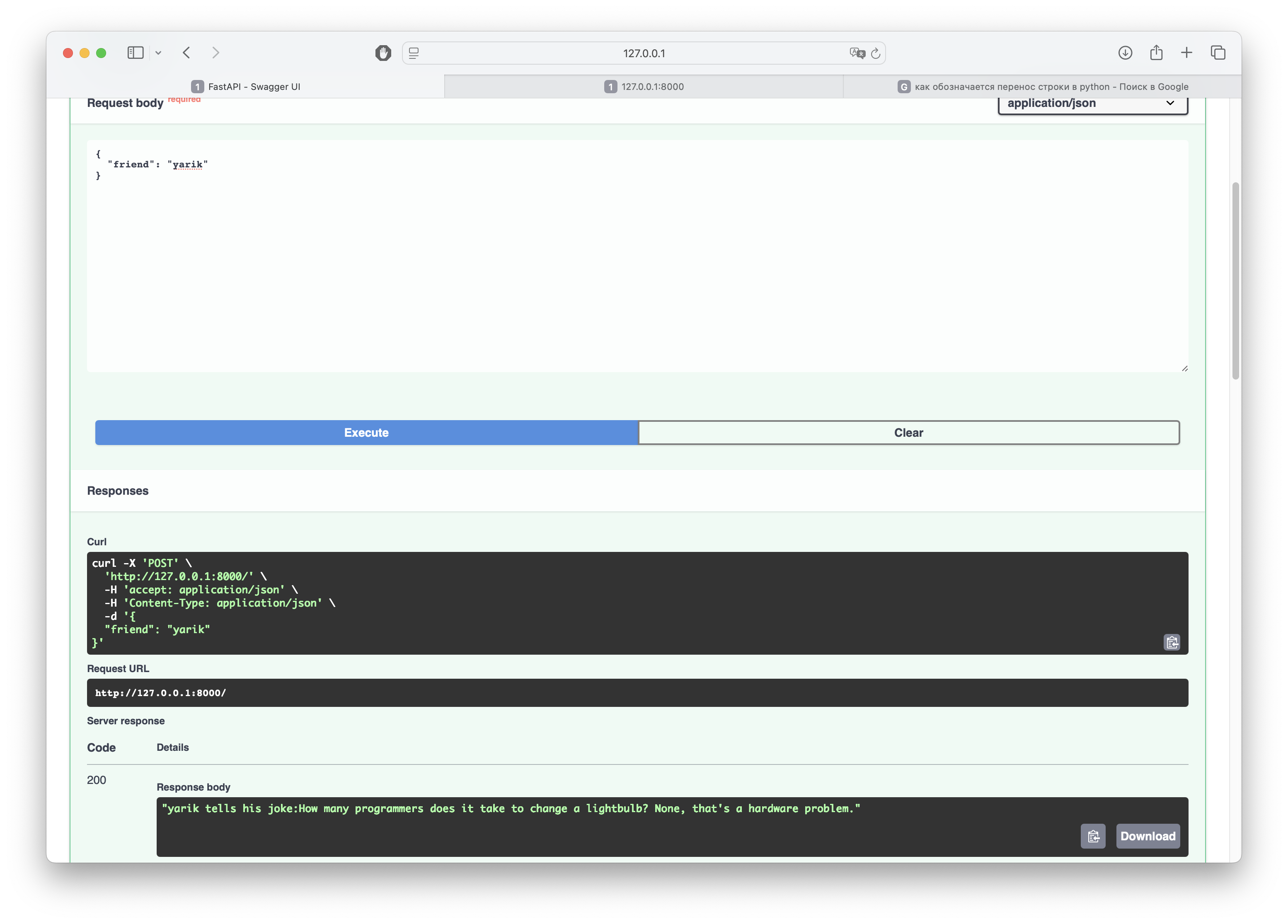
* Добавим количество шуток при выводе:

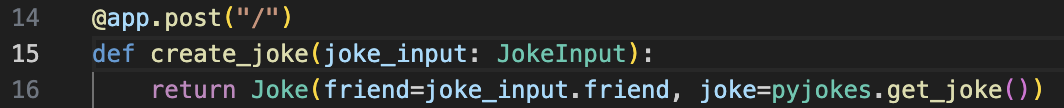
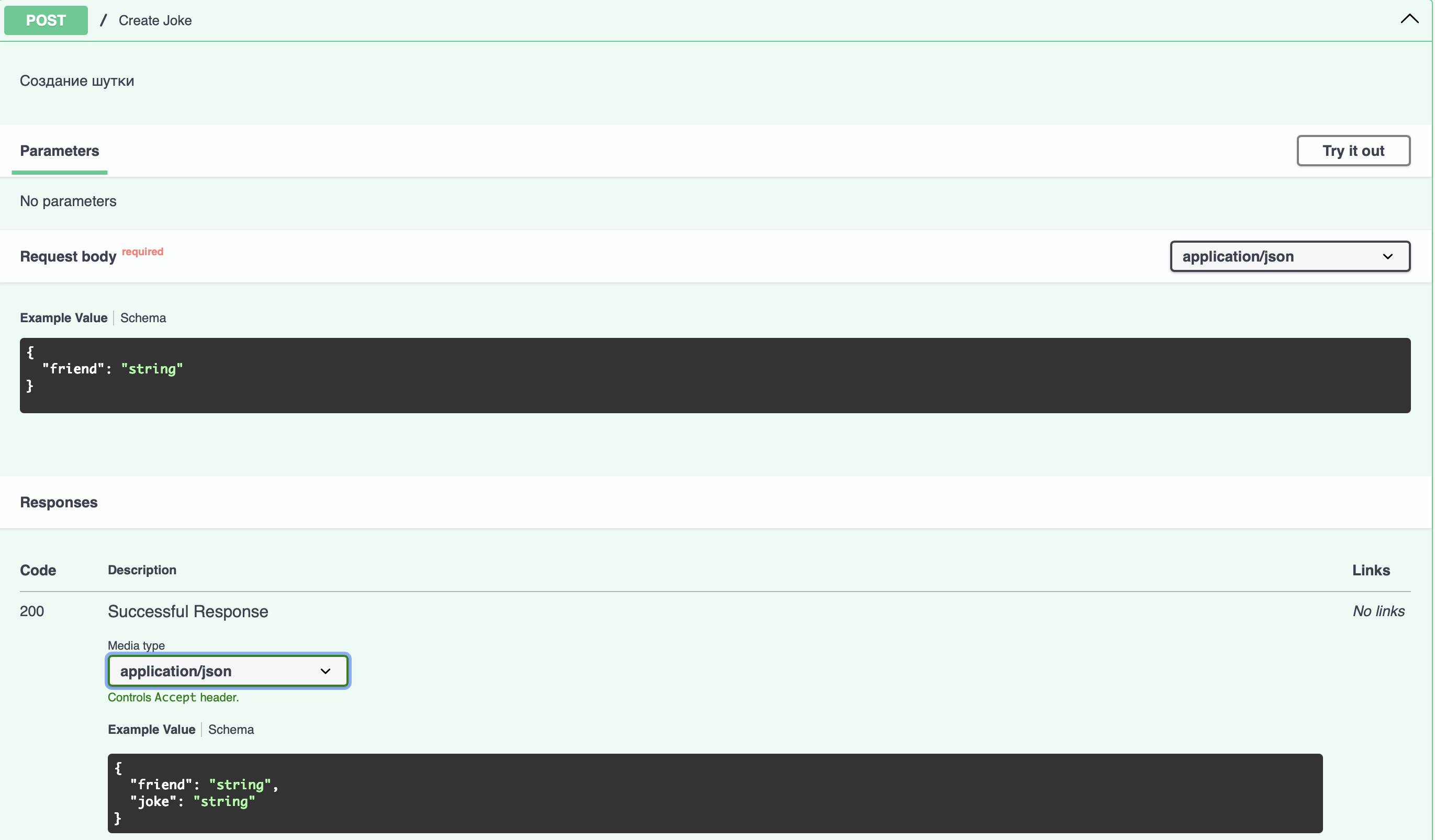
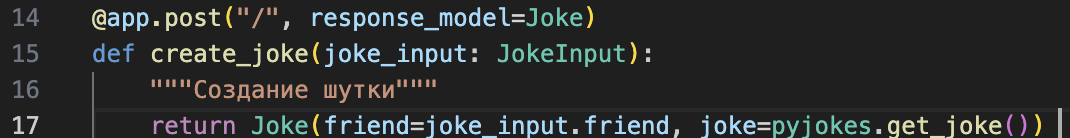


Рассмотрим и другой способ передачи информации в роут. Например, некоторые http запросы поддерживают передачу данных в теле запроса (Body). Создадим новый роут, использую метод POST, и создадим схему тела запроса, которую будет принимать роут для корректной работы. Импортируем из библиотеки pydantic класс BaseModel, как показано на рисунке ниже.

from pydantic import BaseModel

И создадим на его основе схему получения шутки, которую и передадим функции на вход, как показано ниже.

Делаем успешный post запрос

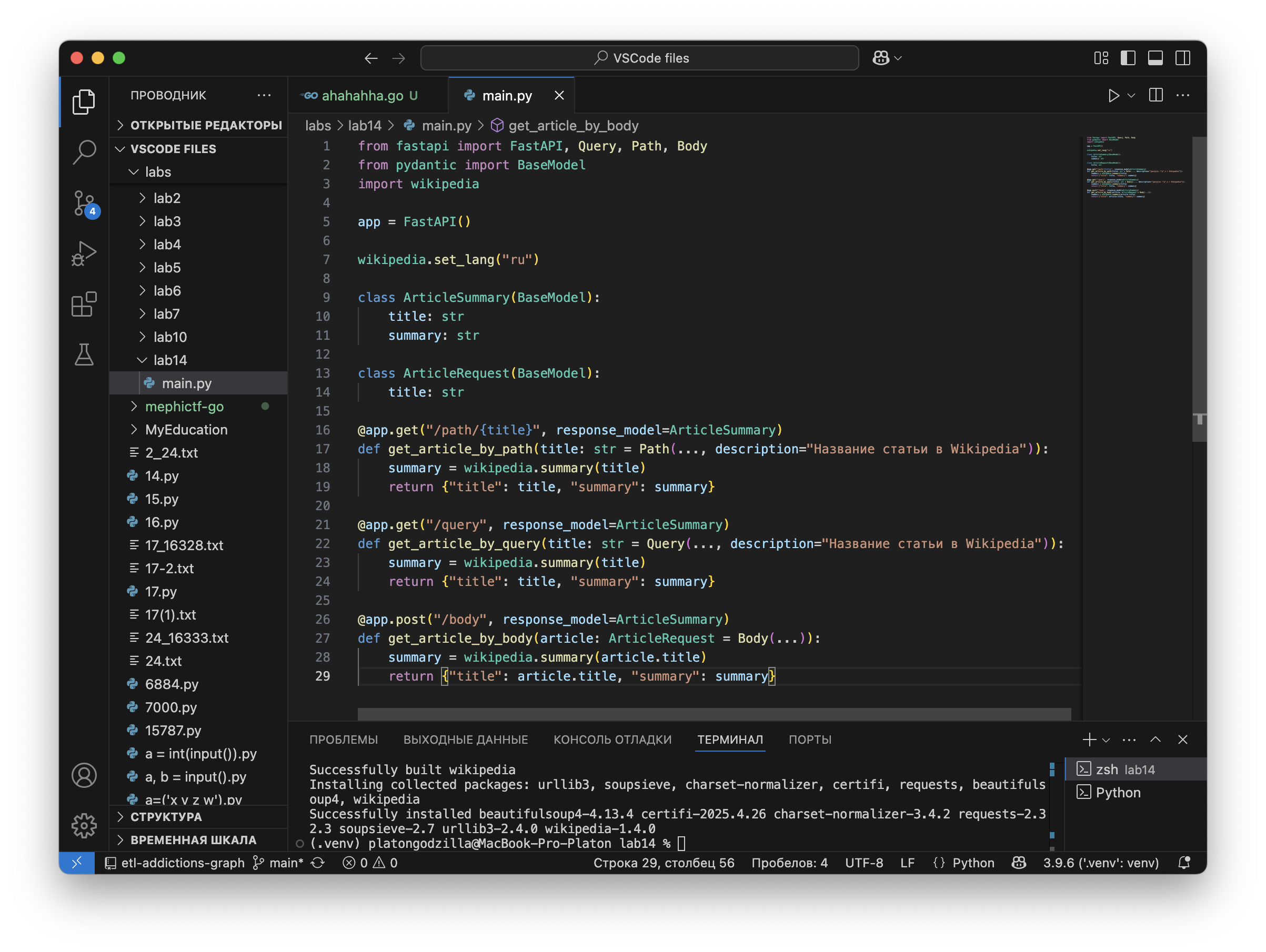
* Сделаем ожидаемый запрос
* Добавим комментарий с описанием роута

Самостоятельное задание:

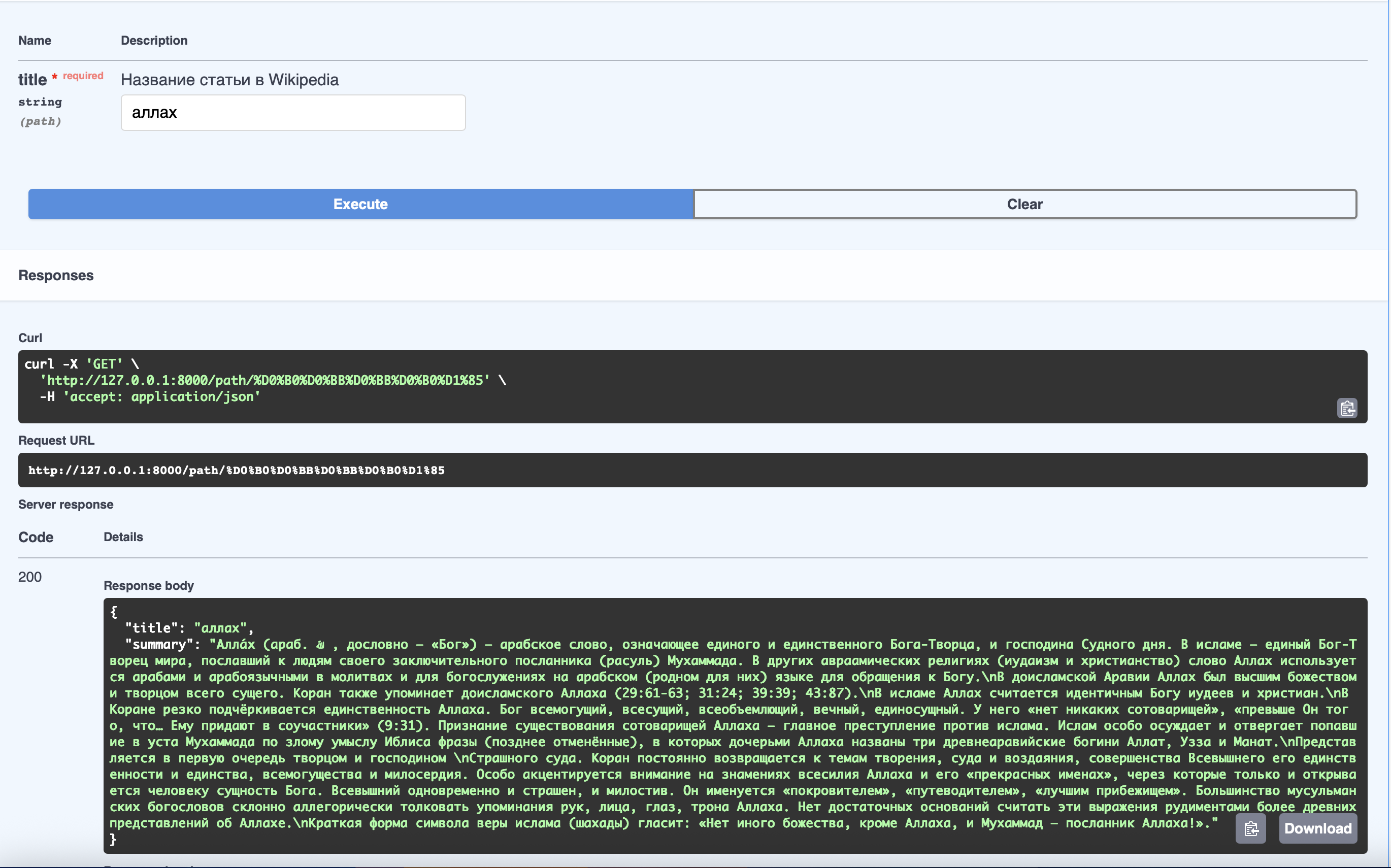
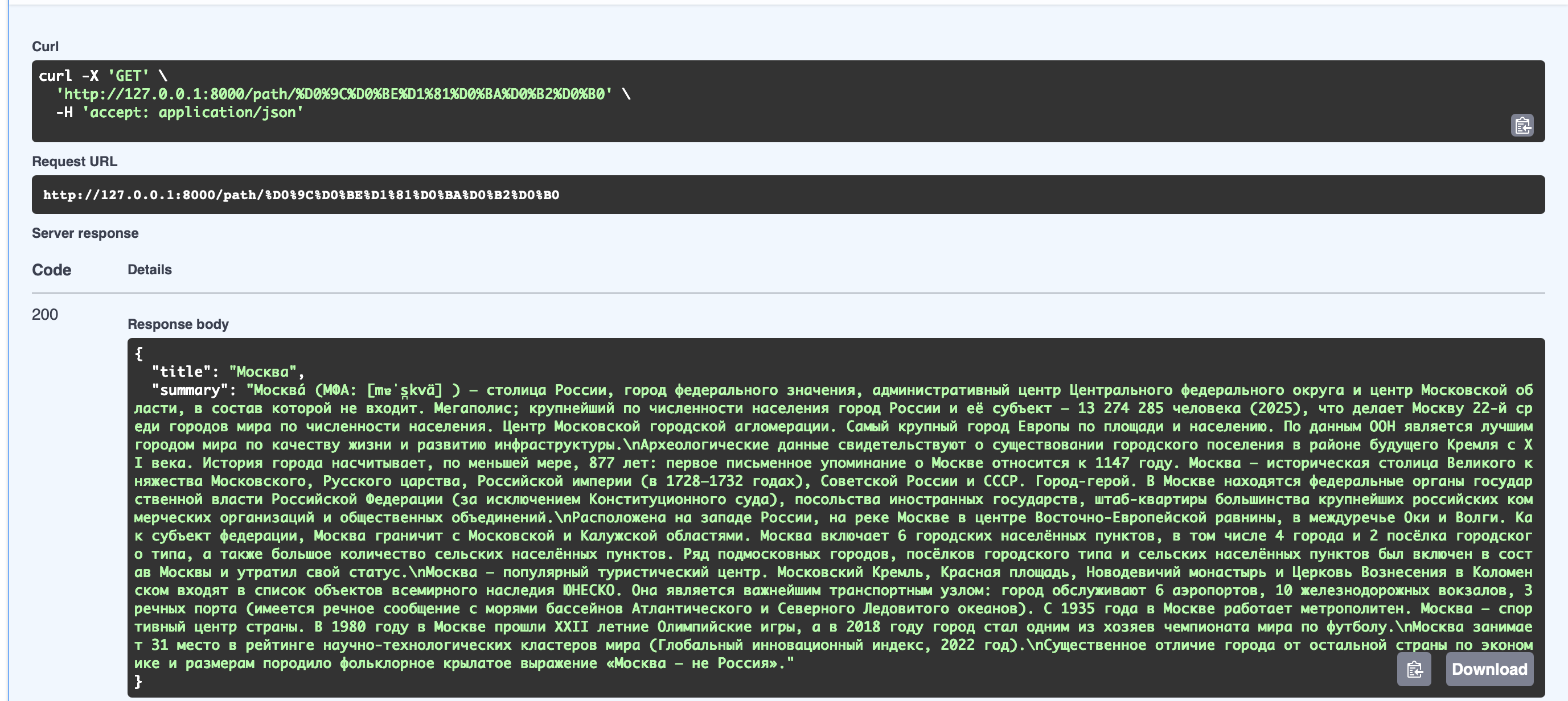
Создать простой REST сервис, использующий библиотеку wikipedia. Создайте 1 роут с параметром path, 1 роут с параметром query, 1 роут с передачей параметров в теле запроса. Все запросы должны возвращаться и валидирироваться по схемам.

Ссылка на документацию по Wikipedia в python

<https://rukovodstvo.net/posts/id_1061/>



1. **Вывод:**

Мы познакомились с основами работы FastAPI