**МИНИСТЕРСТВО 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:

# FastAPI

Выполнил: студент группы БПИ2401

Бамбаев Алтан

Проверил: Мкртчян Грач Маратович

Москва

2025

**Цель** 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/>

**код работы**

from fastapi import FastAPI

from pydantic import BaseModel

import wikipedia

app = FastAPI()

wikipedia.set\_lang("ru")

class PageRequest(BaseModel):

    title: str

@app.get("/summary/{title}")

async def path\_summary(title: str):

    try:

        return {"summary": wikipedia.summary(title)}

    except:

        return {"error": "Статья не найдена"}

@app.get("/search")

async def search(q: str, limit: int = 5):

    return {"results": wikipedia.search(q, results=limit)}

@app.post("/page")

async def body\_page(request: PageRequest):

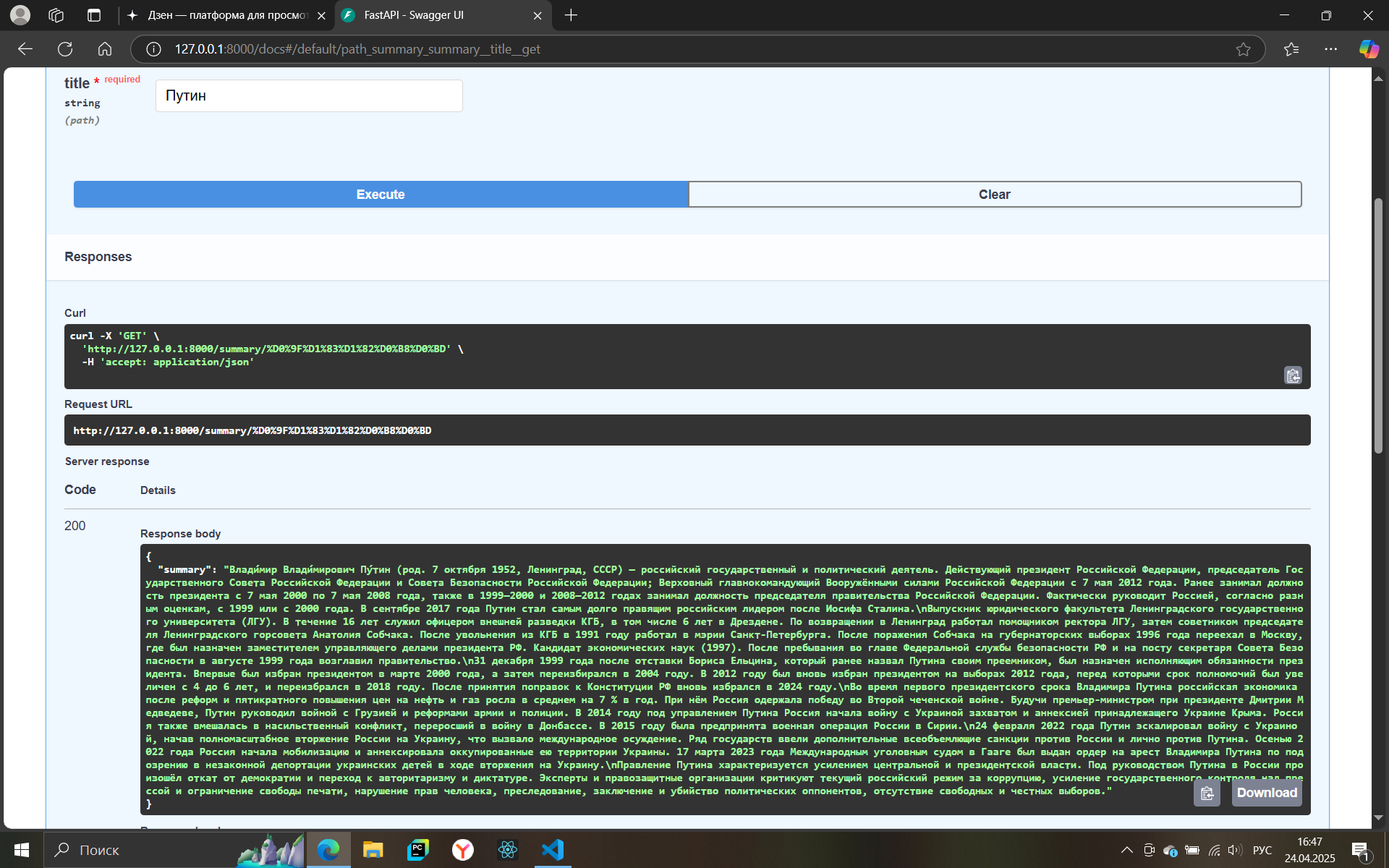
    return {

        "title": request.title,

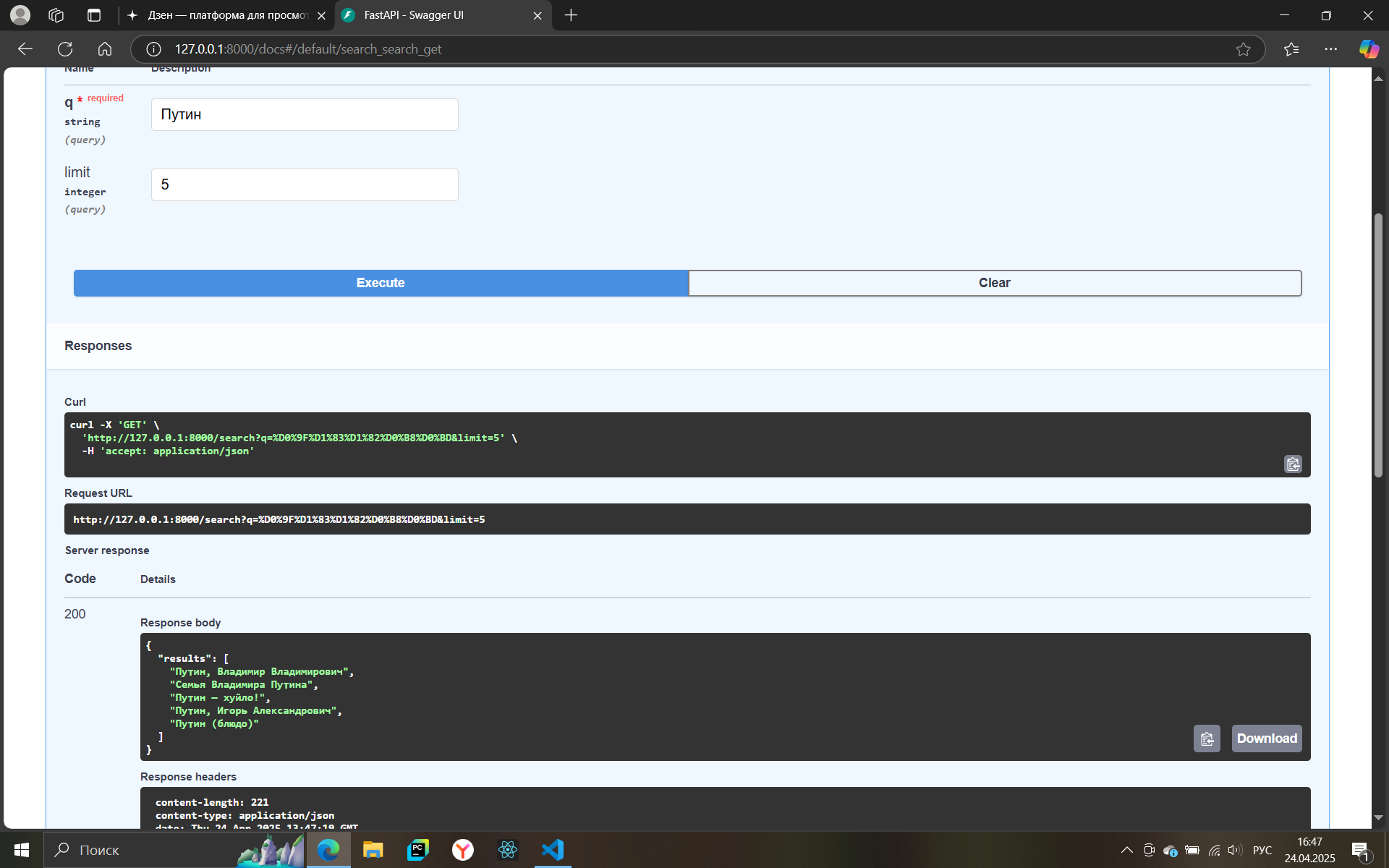
        "url": wikipedia.page(request.title).url

    }

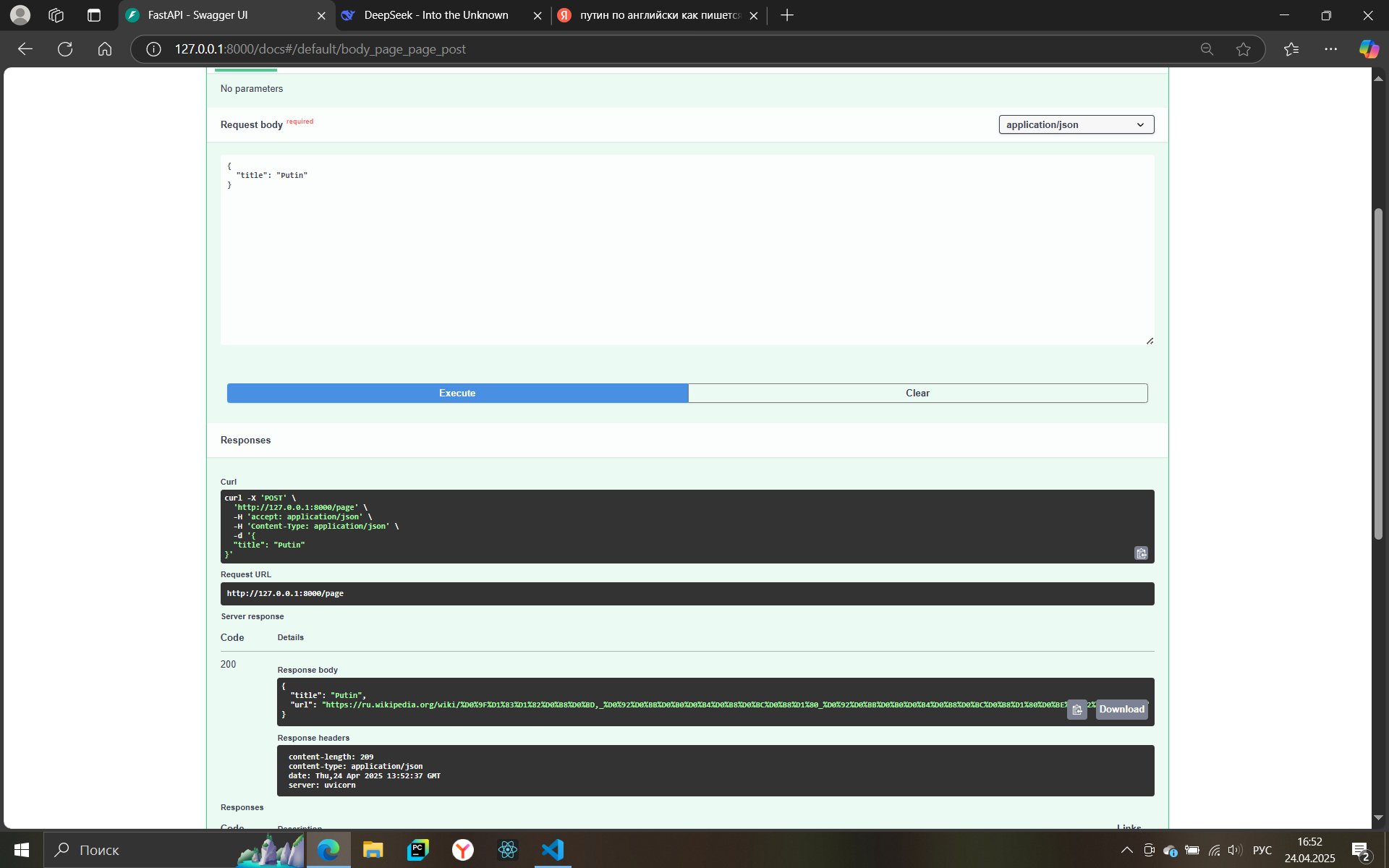
### Возвращает ****краткое описание (summary)**** статьи из Википедии по её названию



1. Ищет статьи википедии по ключевому слову и выдаёт по лимиту



3.принимает название статьи в теле запроса и возвращает название и ссылку на статью



**Вывод: мы прикоснулись к изучению fastapi**