

Міністерство освіти і науки України

Національний технічний університет України „КПІ імені Ігоря Сікорського ”

Факультет інформатики та обчислювальної техніки Кафедра інформатики і програмної інженерії

# ЗВІТ

лабораторної роботи № 6

з курсу «Основи WEB - технологій»

Тема: «Створення власного API на Python (FastAPI).»

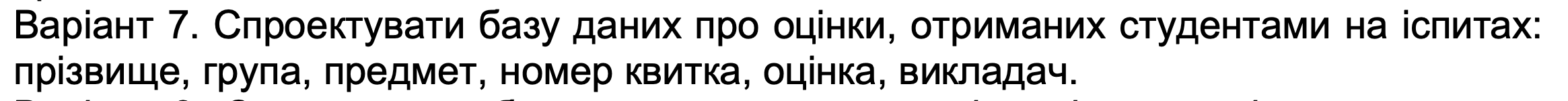
|  |  |
| --- | --- |
| Перевірив:  Викл. Альбрехт Й.О. | Виконав:  Студент ІП-15 Мєшков А.І |

Київ 2024

# Завдання

Створити web-додаток з використанням технології FastAPI, який буде працювати з інформацією відповідно вашому варіанту.

Варіант 7



# Хід роботи

Було програму python з використанням fastapi, pydantic, typing.

**main.py**

*from* fastapi *import* FastAPI, HTTPException

*from* pydantic *import* BaseModel

*from* typing *import* List

app = FastAPI()

class Grade(BaseModel):

id: int

lastName: str

group: str

subject: str

ticketNumber: int

grade: int

professor: str

grades = []

@app.get("/grades", response\_model=List[Grade])

async def get\_all\_grades():

*return* grades

@app.get("/grades/{grade\_id}", response\_model=Grade)

async def get\_grade(grade\_id: int):

*for* grade *in* grades:

*if* grade.id == grade\_id:

*return* grade

*raise* HTTPException(status\_code=404, detail="Grade not found")

@app.post("/grades", response\_model=Grade)

async def create\_grade(grade: Grade):

*for* existing\_grade *in* grades:

*if* existing\_grade.id == grade.id:

*raise* HTTPException(status\_code=400, detail="Grade with this ID already exists")

grades.append(grade)

*return* grade

@app.put("/grades/{grade\_id}", response\_model=Grade)

async def update\_grade(grade\_id: int, updated\_grade: Grade):

*for* i, grade *in* enumerate(grades):

*if* grade.id == grade\_id:

grades[i] = updated\_grade

*return* updated\_grade

*raise* HTTPException(status\_code=404, detail="Grade not found")

@app.delete("/grades/{grade\_id}")

async def delete\_grade(grade\_id: int):

*for* i, grade *in* enumerate(grades):

*if* grade.id == grade\_id:

grades.pop(i)

*return* {"message": f"Grade with ID {grade\_id} deleted successfully"}

*raise* HTTPException(status\_code=404, detail="Grade not found")

# Отримані результати

На рис 6.1-6.6 можна побачити запити.



# Рис. 6.1. Запити

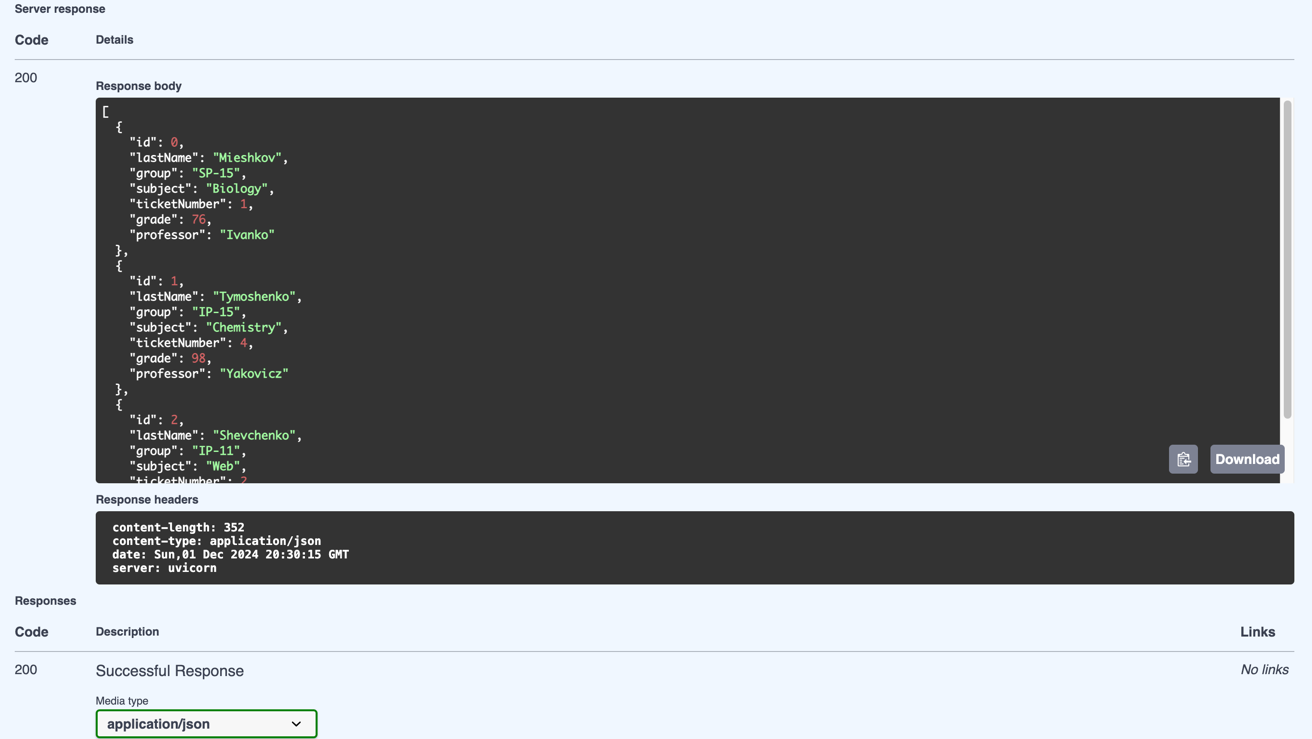
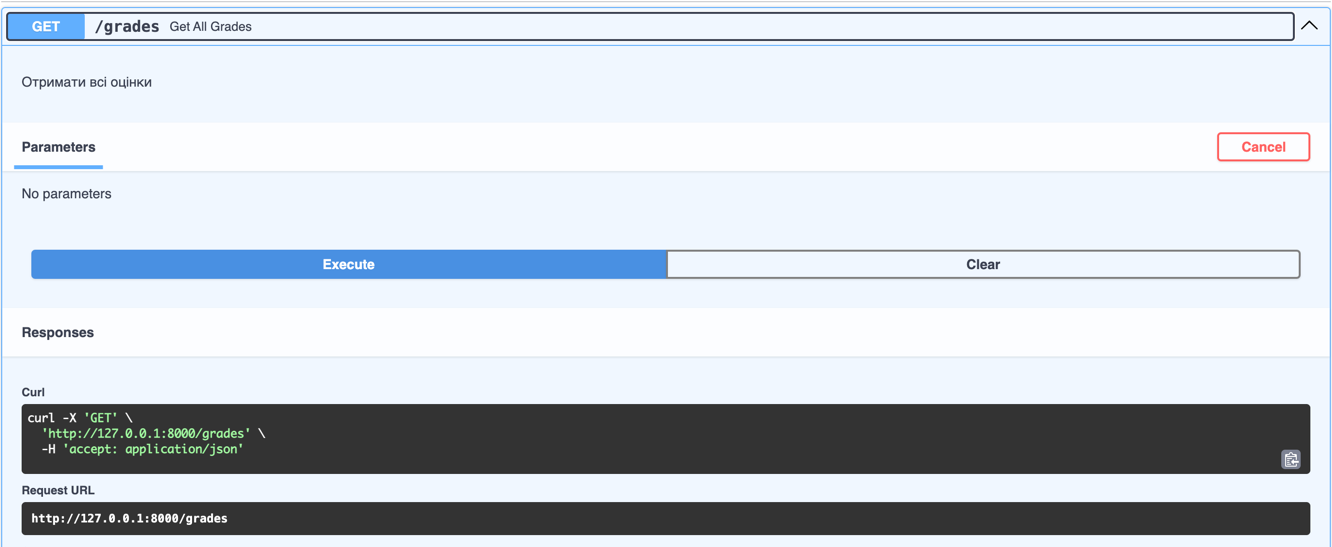


Рис. 6.2. Запит для виведення всіх документів в БД

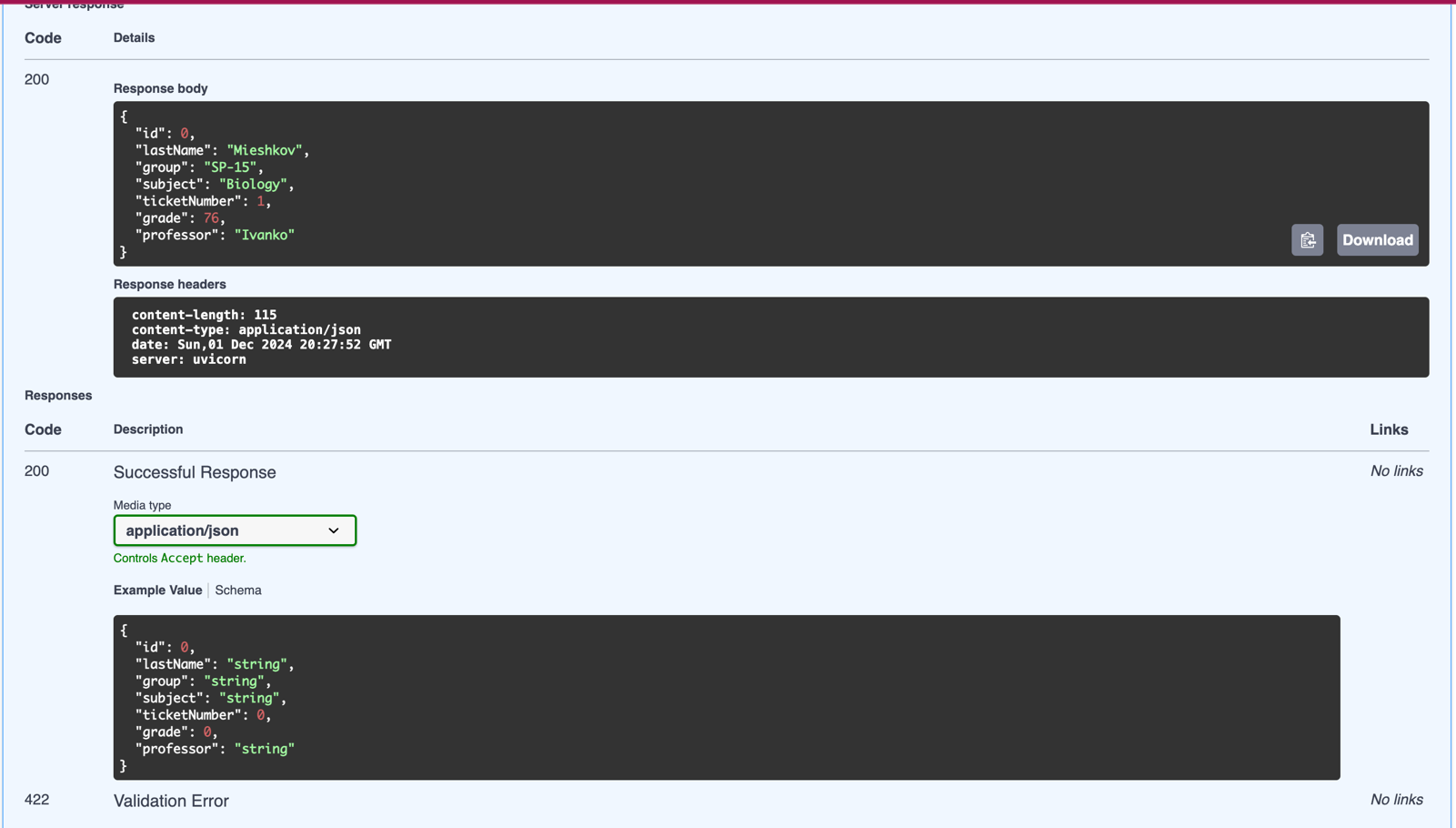
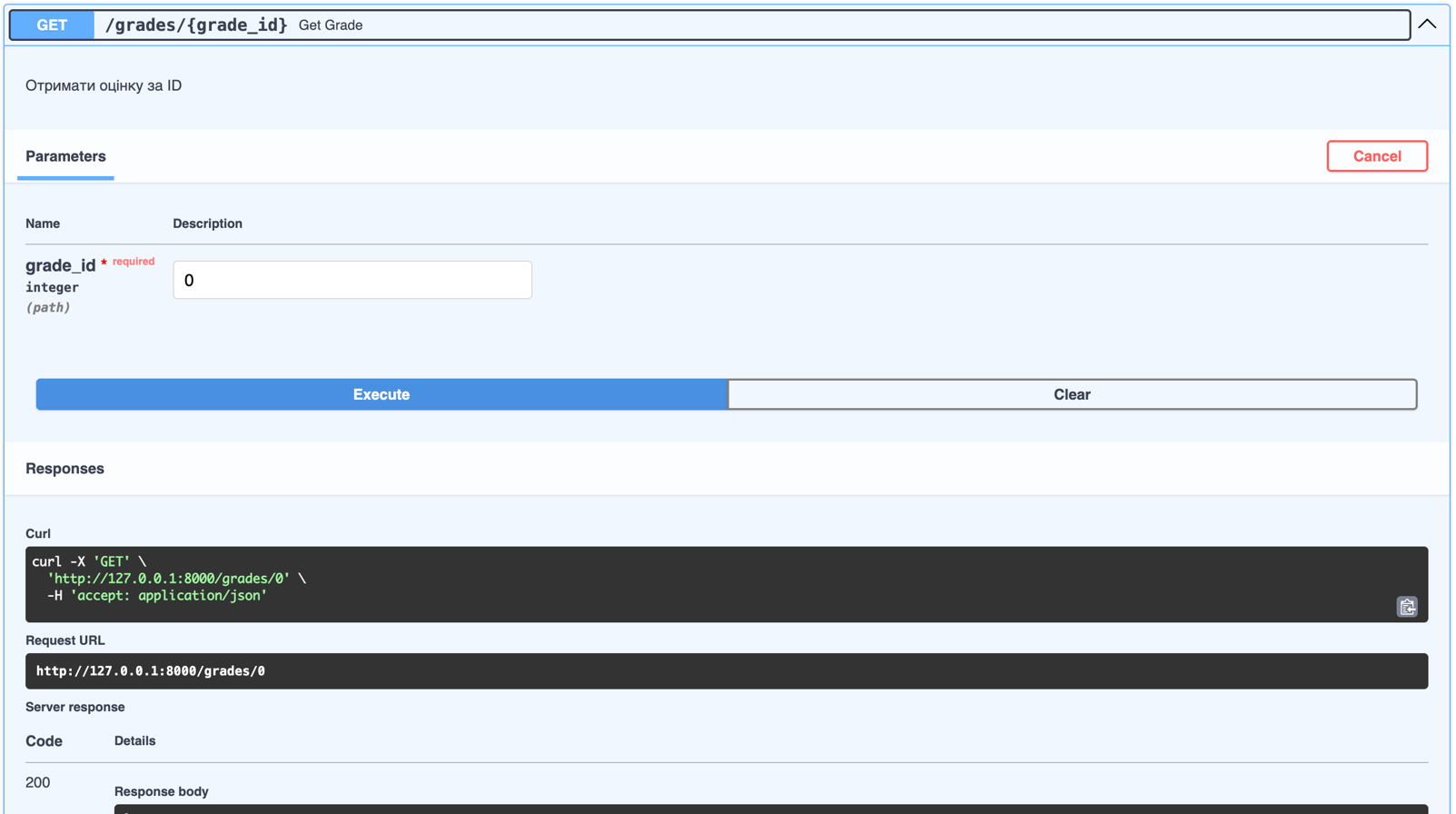


Рис. 6.3. Запит для виведення заданого документу в БД

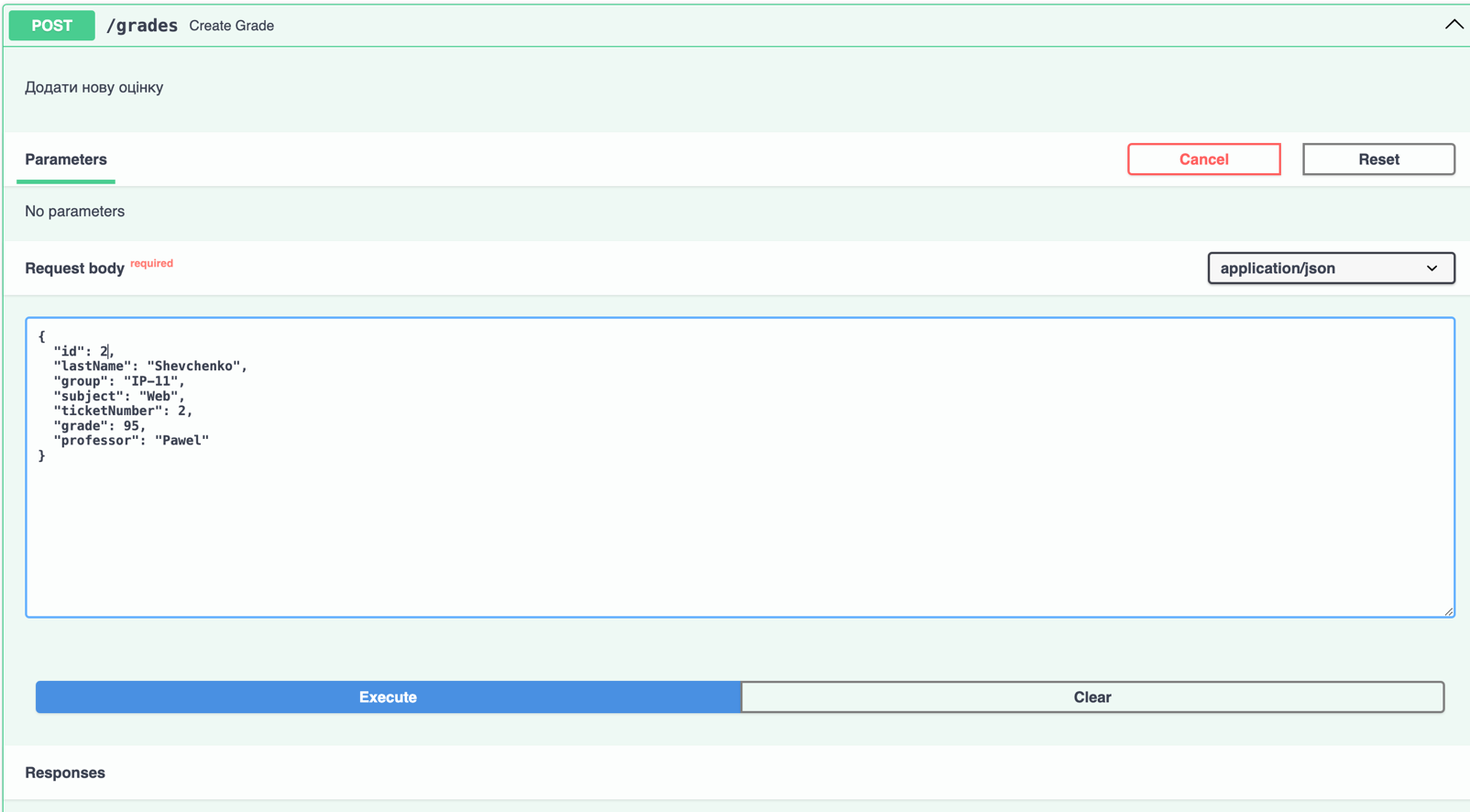


Рис. 6.4. Запит для додавання документу в БД

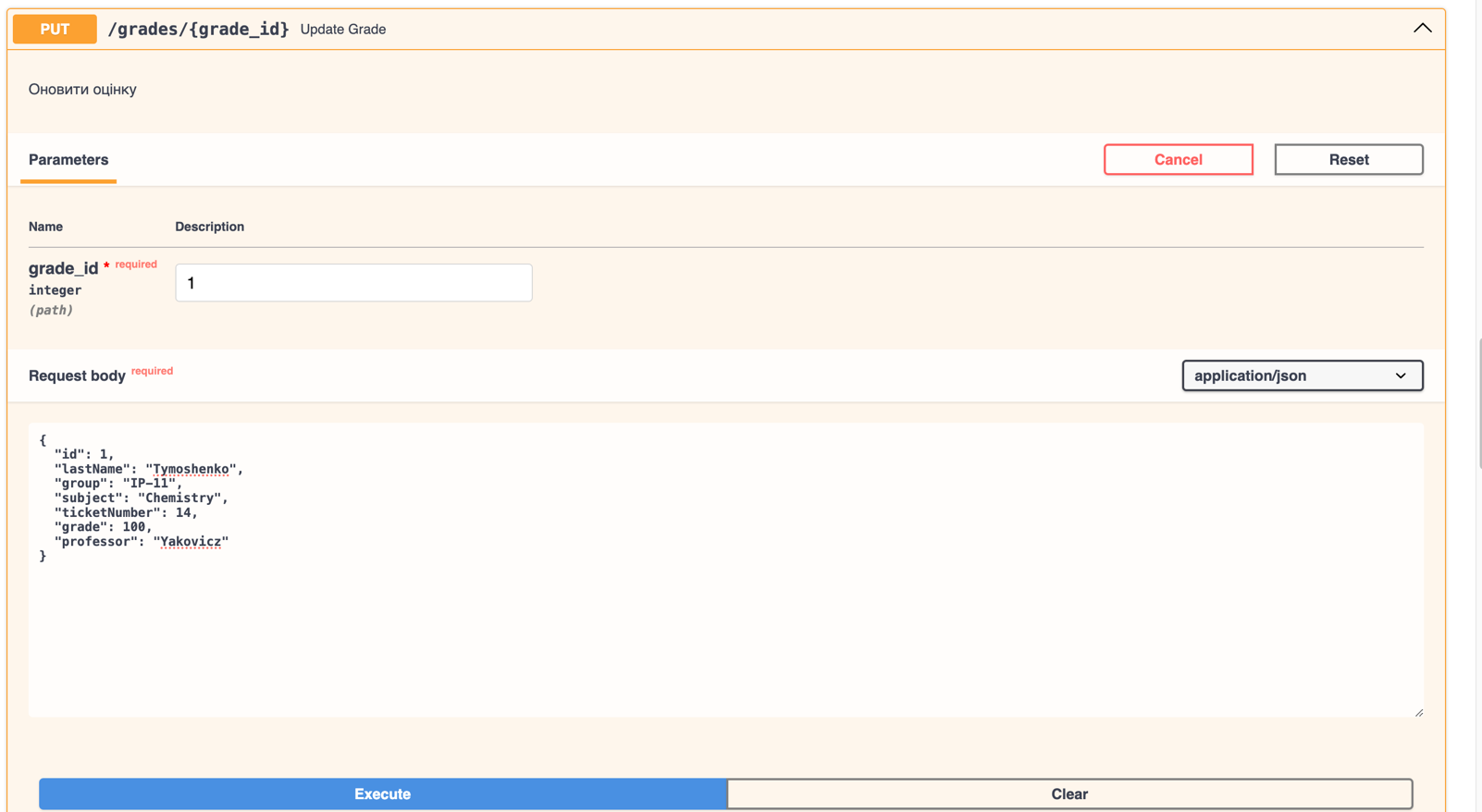
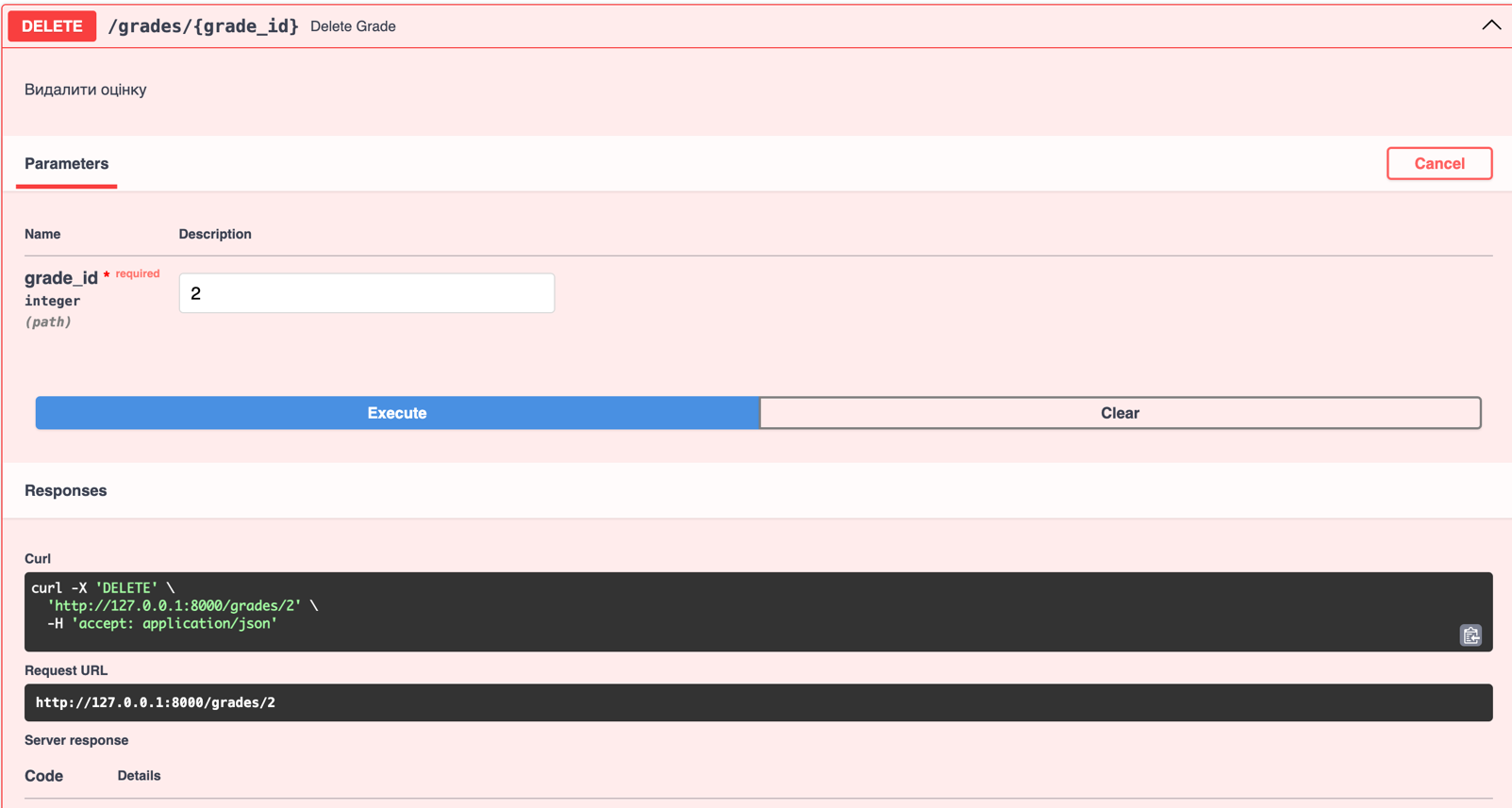


Рис. 6.5. Запит для оновлення документу в БД



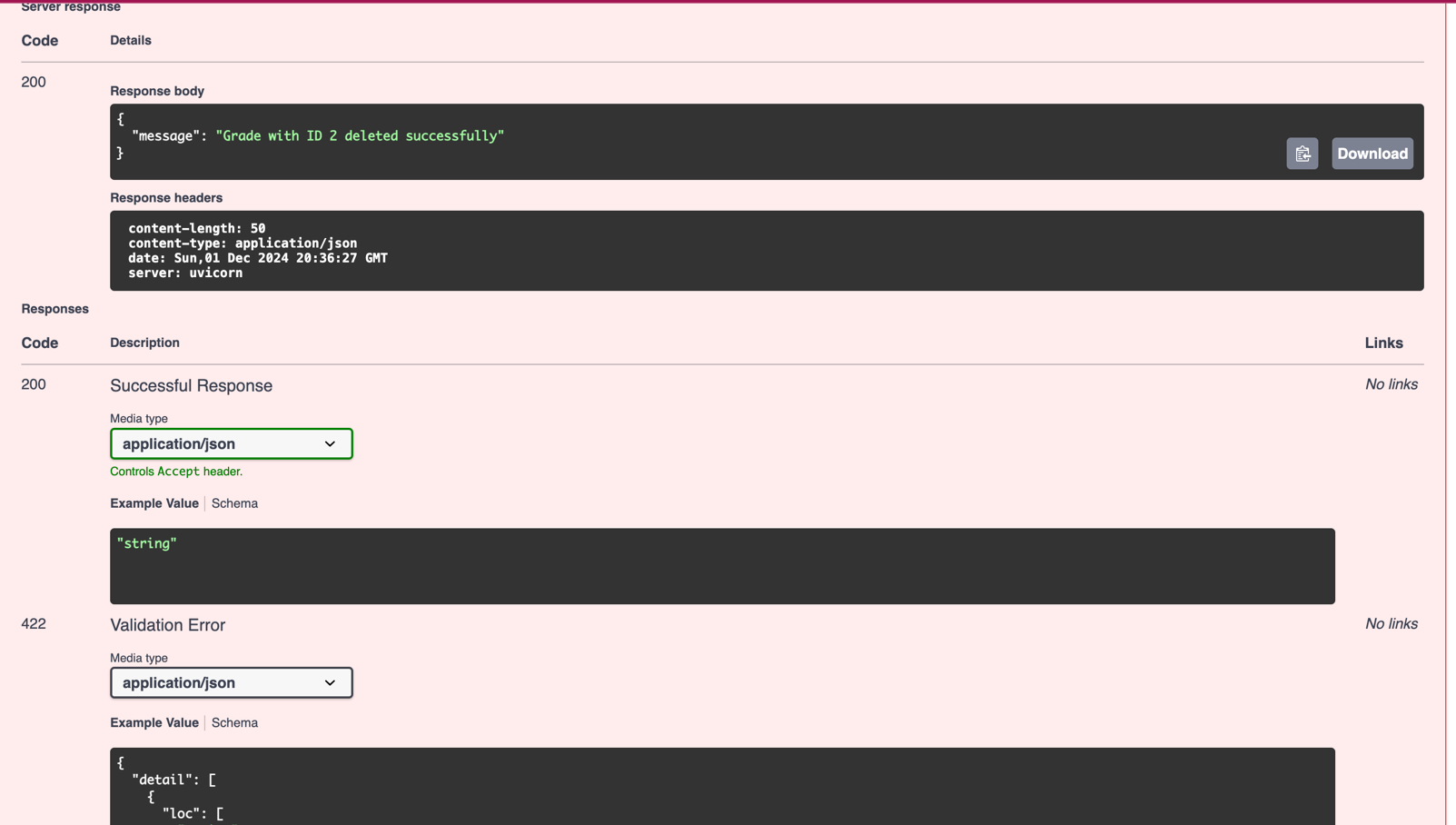


Рис. 6.6 Запит для видалення документу в БД

# ВИСНОВОК

Під час виконання даної лабораторної роботи було створено web-додаток на базі технології FastAPI для реалізації API, що дозволяє працювати з інформацією про оцінки студентів.

Було розроблено схему даних за допомогою Pydantic, створено маршрути для CRUD-операцій (додавання, редагування, видалення та перегляду даних), а також протестовано функціональність API.

В результаті:

* Реалізовано ефективну модель зберігання даних у пам'яті додатка.
* Отримано практичний досвід роботи з FastAPI, Pydantic та тестуванням HTTP-запитів.
* Набуто розуміння основ розробки RESTful API, яке може бути масштабоване на складніші проекти з використанням баз даних.

Додаток виконує поставлені завдання відповідно до варіанту роботи, демонструючи основні можливості FastAPI для створення сучасних веб-сервісів.