FINÁLNÍ PROJEKT
č.1

Autor: Vojtěch Machytka
Datum: 12.12.2024

OBSAH

ZADÁNÍ

Cílem finálního projektu je otestovat funkčnost aplikace, která slouží k manipulaci s daty o studentech. Aplikace má rozhraní REST-API, které umožňuje vytvoření, smazání a získání dat..

Přístupové údaje:

Databáze	database: qa_demo Host: aws.connect.psdb.cloud Username: Password:
REST-API	http://108.143.193.45:8080/api/v1/students/

TESTOVACÍ SCÉNÁŘE

1. FUNKCE GET

1. Získat data o existujícím studentovi

Testovací data:
ID studenta = 1127

V MySQLWorkbench:
SELECT * FROM student where id=1127;
'1127','19','jc34@yourmail.com','John','SCOTT'

Steps:
1.) GET http://108.143.193.45:8080/api/v1/students/1127
2.) Press Send button
Expected result:
1.) status code 200 OK
2.) Response

```
{
   "id": 1127,
   "firstName": "John",
   "lastName": "SCOTT",
   "email": "jc34@yourmail.com",
   "age": 19
}
```

2. Získat data o neexistujícím studentovi

Testovací data:
ID studenta = 446

V MySQLWorkbench:
SELECT * FROM student where id=446;
0 rows returned

Steps:
1.) GET http://108.143.193.45:8080/api/v1/students/446
2.) Press Send button
Expected result:
1.) status code 404 Not found

2. FUNKCE POST

1. Založit úspěšně nového studenta

Steps:
Zvolím POST
Request body JSON:
{
“firstName": "Carl",
"lastName": "Bananaman",
"email": "cbnman@yourmail.com",
"age": 42
}
Expected results:
1.) status code 201 created

2.) Response:
{
“id”: 1450,
“firstName": "Carl",
"lastName": "Bananaman",
"email": "cbnman@yourmail.com",
"age": 42
}

2. Založit studenta staršího než 150 let

Steps:
Zvolím POST
Request body JSON:
{
“firstName": "Frankie",
"lastName": “Oldman”,
"email": "oldfrankie@aol.com",
"age": 238
}
Expected results:
1.) status code 400 BAD REQUEST

3. Založit studenta s neplatným formátem emailu

Steps:
Zvolím POST
Request body JSON:
{
“firstName": "John",
"lastName": “Bolito”,
"email": “johnjohn.aol.com”,
"age": 23
}
Expected results:
1.) status code 400 BAD REQUEST

4. Založit studenta pouze s křestním jménem

Steps:
Zvolím POST
Request body JSON:
{
“firstName": "Wong",
"lastName": “”,
"email": “mrwong@gmail.com”,
"age": 19
}
Expected results:
1.) status code 400 BAD REQUEST

5. Založit studenta s číslem místo příjmení

Steps:
Zvolím POST
Request body JSON:
{
“firstName": "Karel",
"lastName": “4”,
"email": “otecvlasti@praha.cz”,
"age": 52
}
Expected results:
1.) status code 400 BAD REQUEST

6. Založit studenta bez uvedení věku

Steps:
Zvolím POST
Request body JSON:
{
"firstName": "Ruda",
"lastName": "Rudy",
"email": "rudrud@nope.se",
"age":
}
Expected results:
1.) status code 400 BAD REQUEST

3. FUNKCE DELETE

1. Smazat existujícího studenta

Steps:
1.) DELETE http://108.143.193.45:8080/api/v1/students/1450
2.) Press Send button
Expected results:
1.) status code 200 OK

2. Smazat neexistujícího studenta

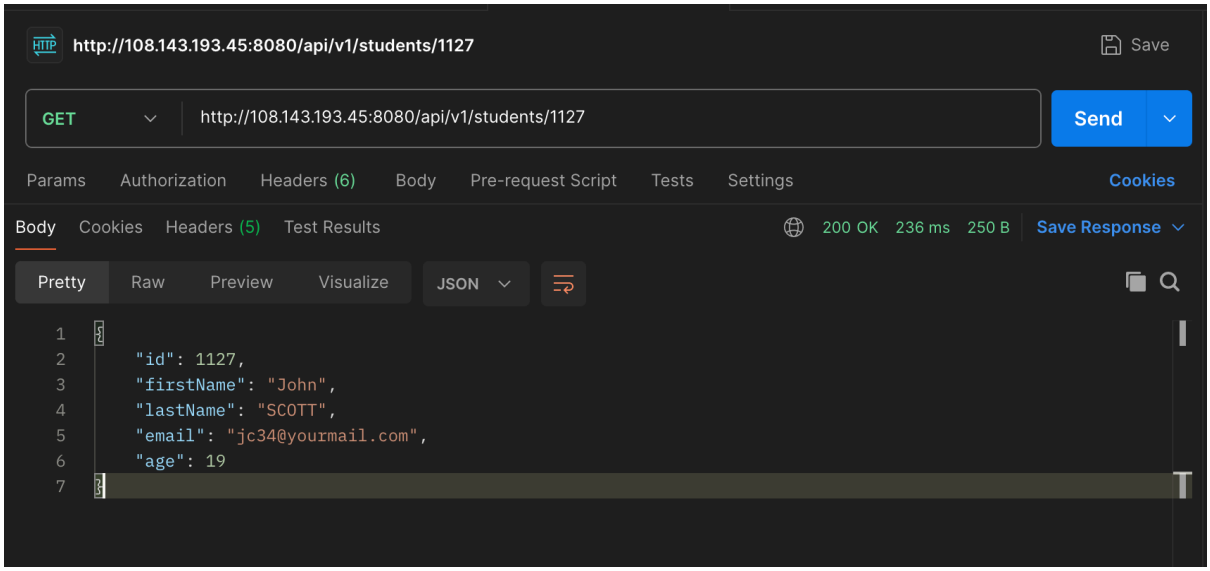
Steps:
1.) DELETE http://108.143.193.45:8080/api/v1/students/1450
2.) Press Send button
Expected results:
1.) status code 404 Not found

EXEKUCE TESTŮ

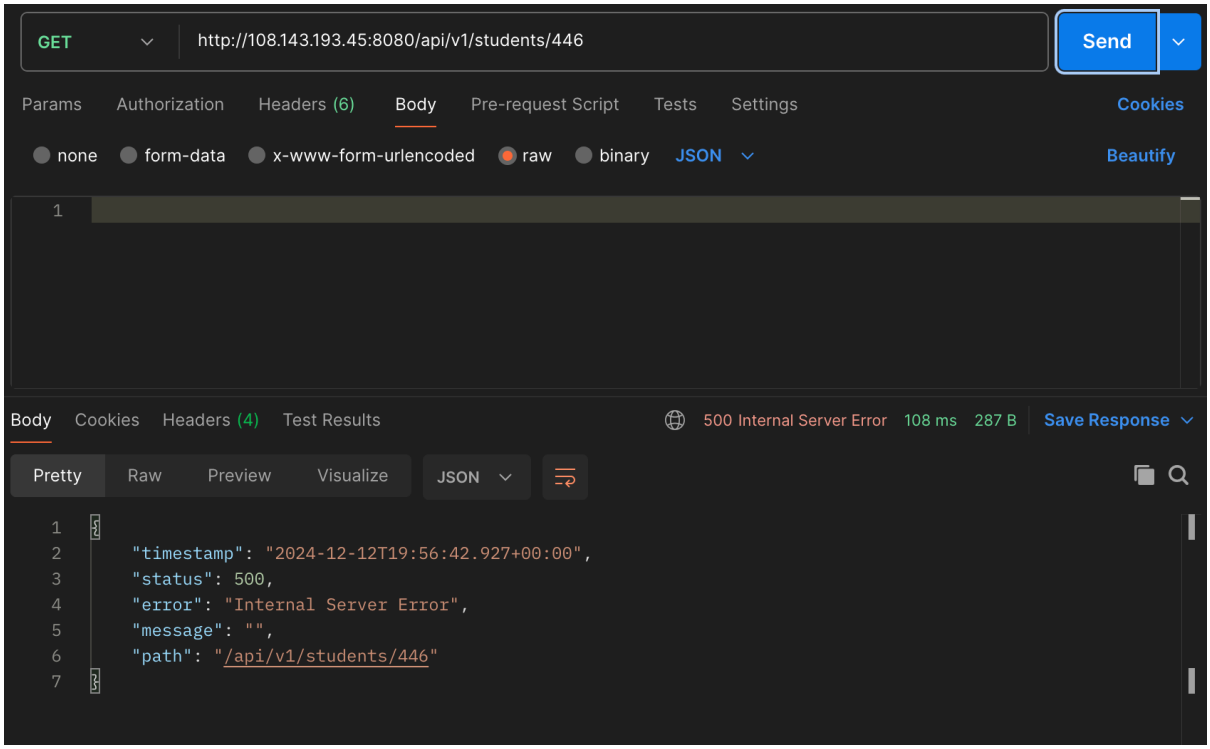
Testovací scénáře jsem provedl, přikládám výsledky testů.

1. FUNKCE GET

1. *Získat data o existujícím studentovi*

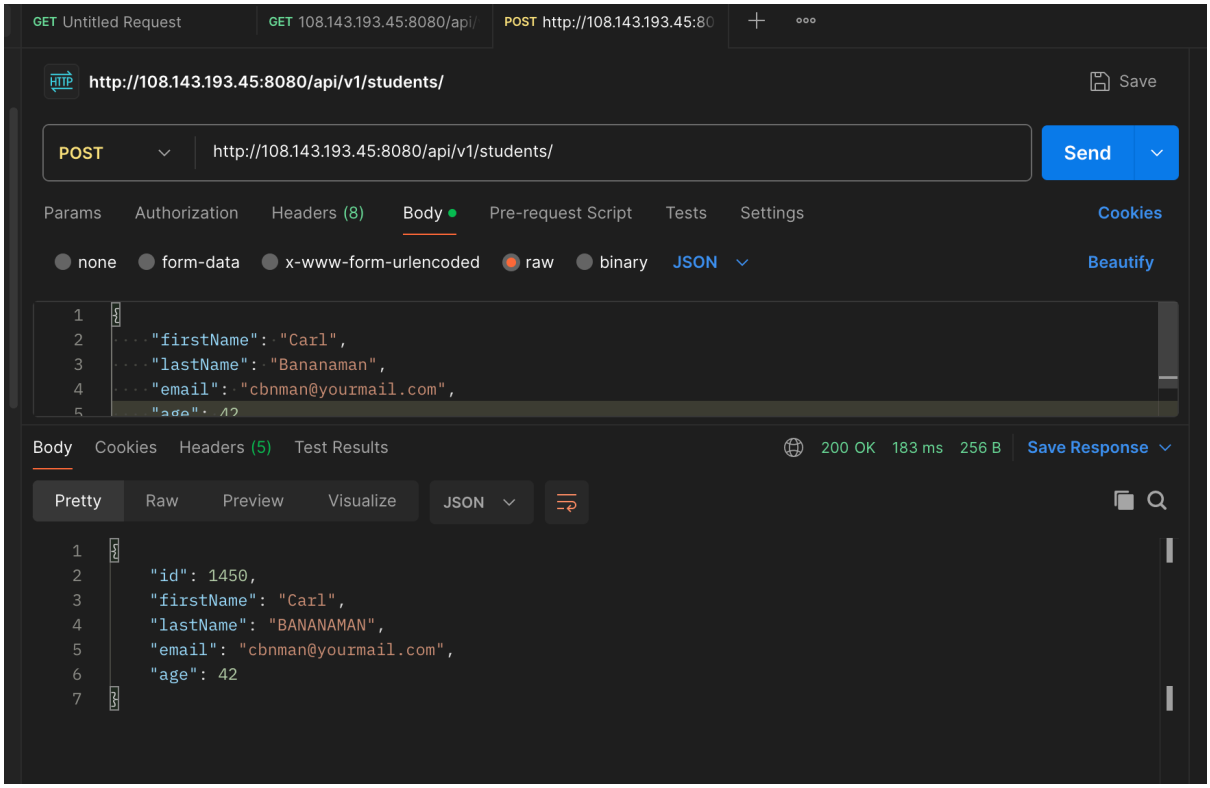


2. Získat data o neexistujícím studentovi

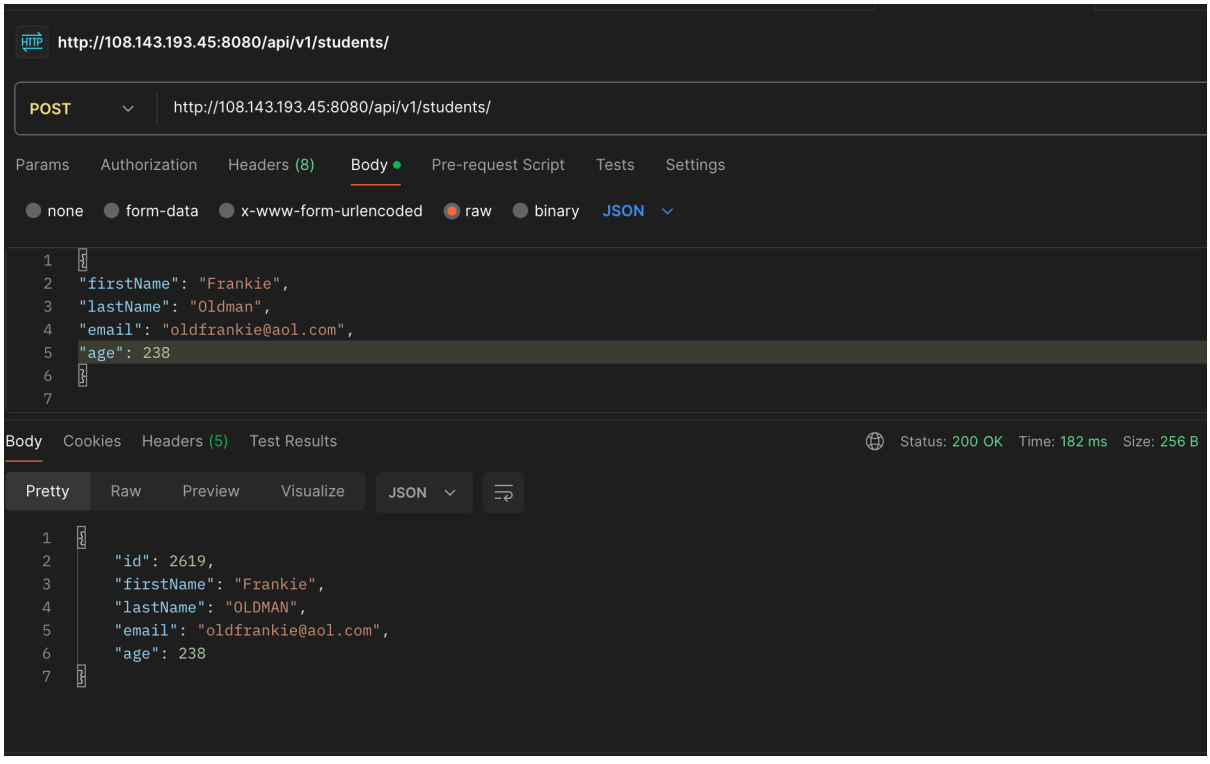


1. FUNKCE POST

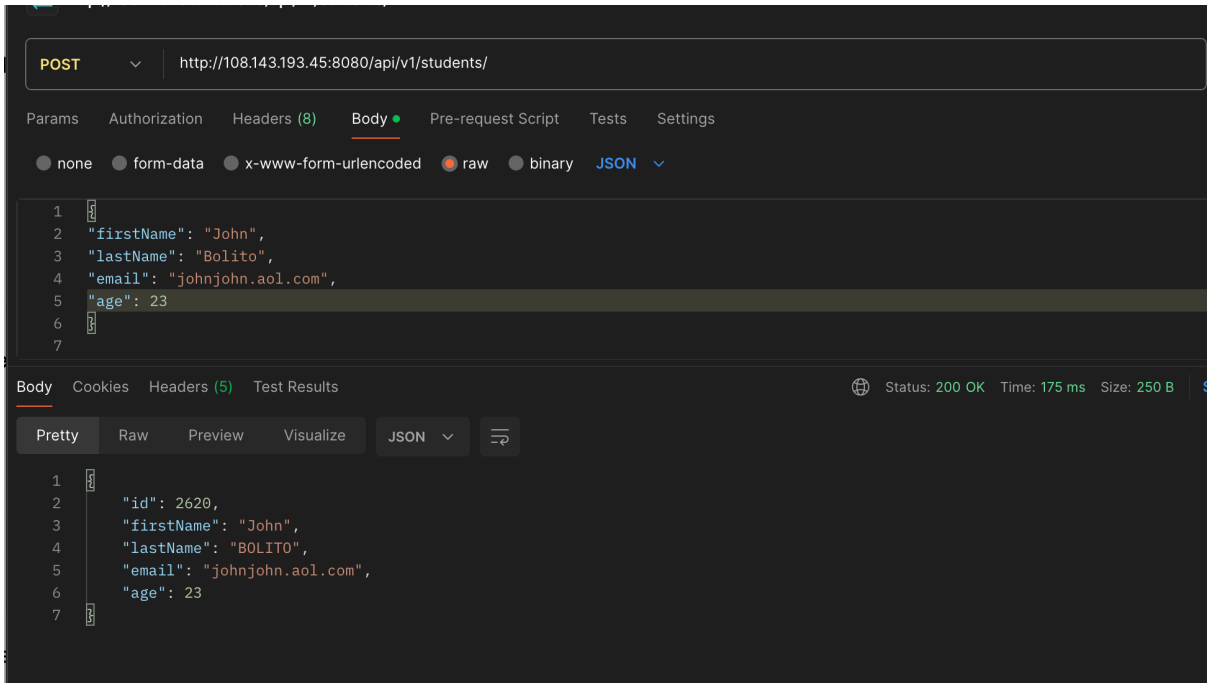
1. Založit úspěšně nového studenta



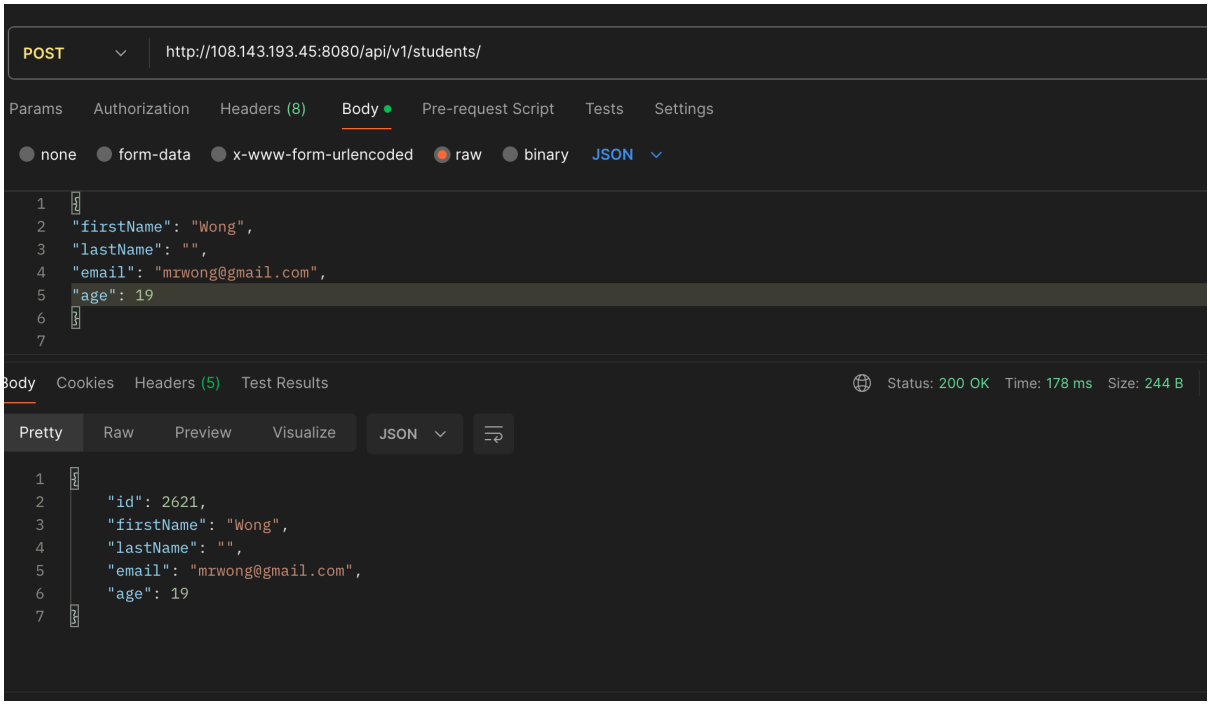
2.Založit studenta staršího než 150 let



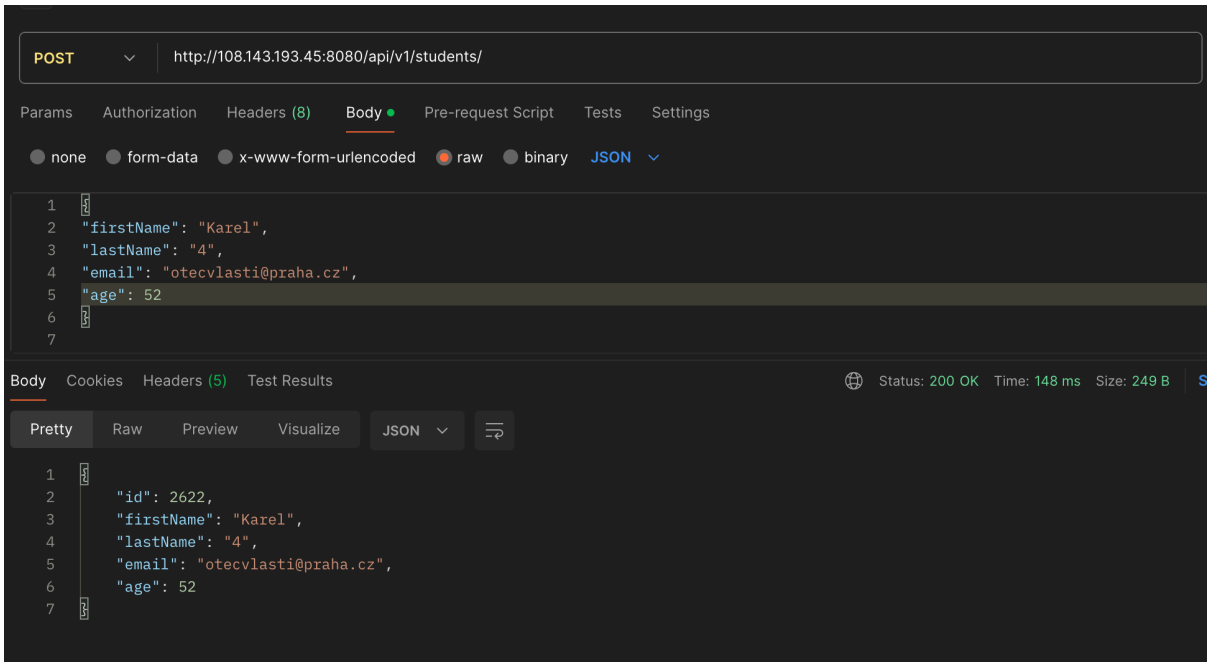
3. Založit studenta s neplatným formátem emailu



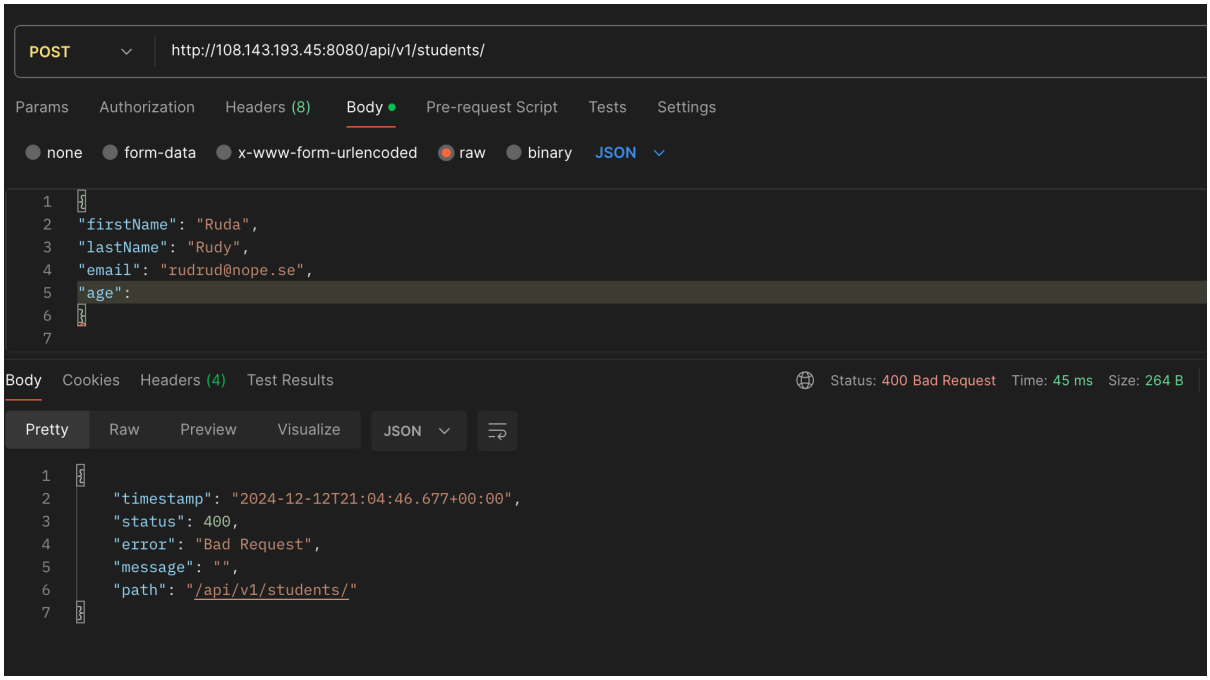
4. Založit studenta pouze s křestním jménem



5. Založit studenta s číslem místo příjmení

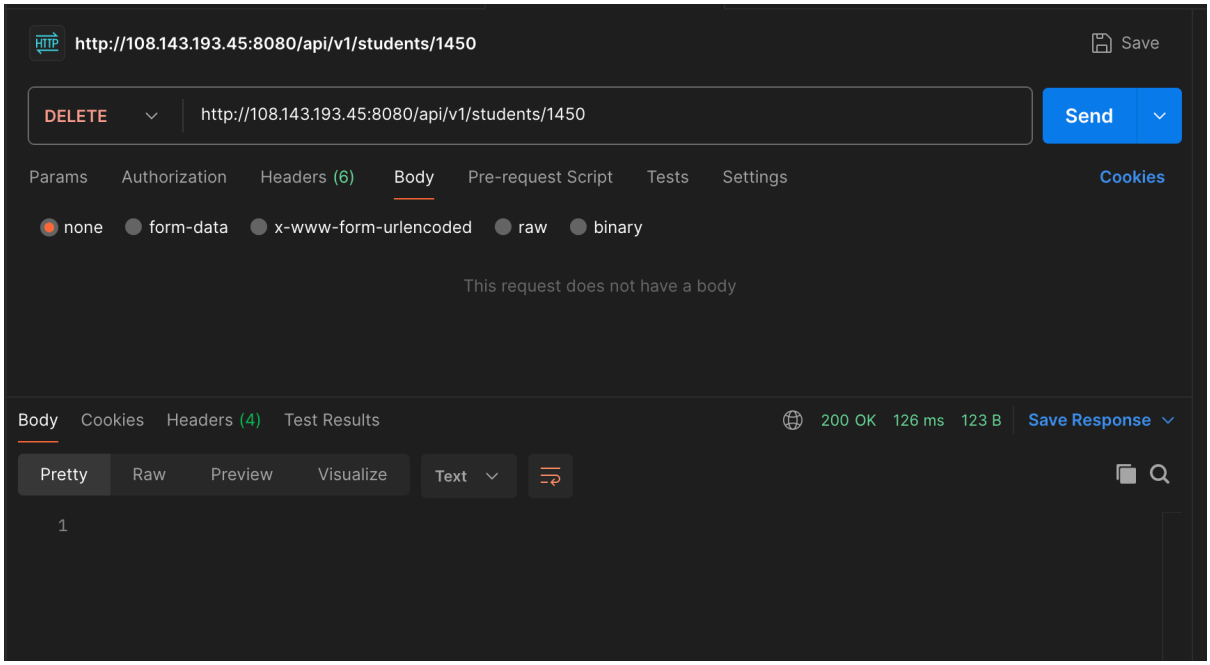


6. Založit studenta bez uvedení věku

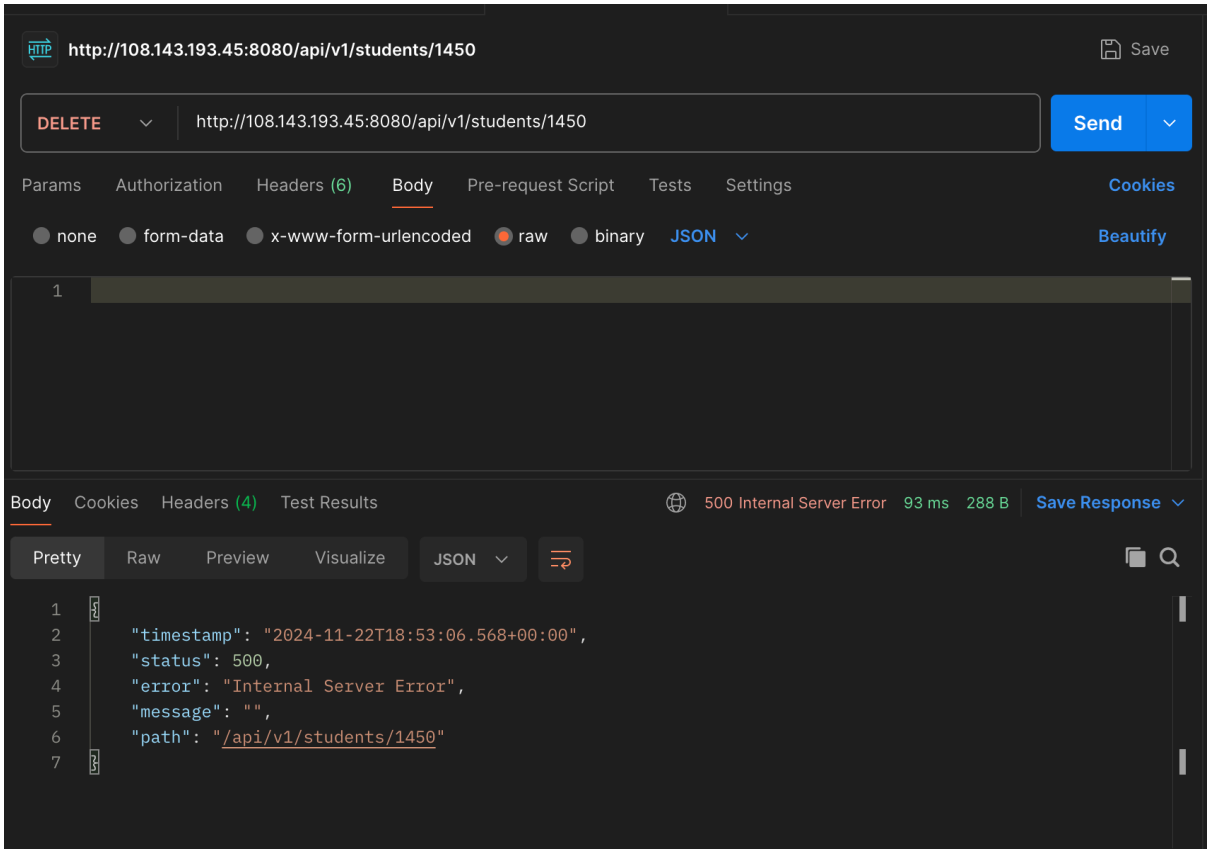


3. FUNKCE DELETE

1. Smazat existujícího studenta



2. Smazat neexistujícího studenta



BUG REPORT

Na základě provedených scénářů jsem objevil uvedené chyby aplikace:

1. *U založení nového studenta je odpověď status 200 OK místo požadovaného 201 Created.*
2. *U funkce get pro neexistující studenta je odpověď status 500 místo požadovaného 404 Not found.*
3. *U založení nového studenta staršího než 150 let status 200 OK místo statusu 400 Bad request.*
4. *U založení nového studenta s neplatným formátem emailu status 200 OK místo statusu 400 Bad request.*
5. *U založení nového studenta bez lastName status 200 OK místo statusu 400 Bad request.*
6. *U založení nového studenta číslem místo příjemní status 200 OK místo statusu 400 Bad request.*
7. *U pokusu o smazání neexistující studenta je odpověď status 500 místo statusu 404 Not found*