FINÁLNÍ PROJEKT   
 č.1



Autor: Lukáš Kovář  
Datum: 25.6.2024

Obsah

[ZADÁNÍ 4](#_Toc170499509)

[Přístupové údaje: 4](#_Toc170499510)

[Poznámky: 4](#_Toc170499511)

[1. TESTOVACÍ SCÉNÁŘ 4](#_Toc170499512)

[1.1. Test the Get method 4](#_Toc170499513)

[1.2. EXEKUCE TESTU 1. 4](#_Toc170499514)

[1.3. BUG REPORT*:* 5](#_Toc170499515)

[2. TESTOVACÍ SCÉNÁŘ 5](#_Toc170499516)

[2.1. Test the Get method 5](#_Toc170499517)

[2.2. EXEKUCE TESTU 2. 5](#_Toc170499518)

[3.2. BUG REPORT*:* 6](#_Toc170499519)

[3. TESTOVACÍ SCÉNÁŘ 6](#_Toc170499520)

[3.1. Test the POST method 6](#_Toc170499521)

[3.2. EXEKUCE TESTU 3. 7](#_Toc170499522)

[3.3. BUG REPORT: 8](#_Toc170499523)

[4. TESTOVACÍ SCÉNÁŘ 8](#_Toc170499524)

[4.1. Test the POST method 8](#_Toc170499525)

[4.2. EXEKUCE TESTU 4. 8](#_Toc170499526)

[4.3.3 BUG REPORT: 9](#_Toc170499527)

[5. TESTOVACÍ SCÉNÁŘ 9](#_Toc170499528)

[5.1. Test the POST method 9](#_Toc170499529)

[5.2. EXEKUCE TESTU 5. 9](#_Toc170499530)

[5.3. BUG REPORT: 10](#_Toc170499531)

[6. TESTOVACÍ SCÉNÁŘ 10](#_Toc170499532)

[6.1. Test the POST method 10](#_Toc170499533)

[6.2. EXEKUCE TESTU 6. 10](#_Toc170499534)

[6.3. BUG REPORT: 11](#_Toc170499535)

[7. TESTOVACÍ SCÉNÁŘ 11](#_Toc170499536)

[7.1 Test the DELETE method 11](#_Toc170499537)

[7.2. EXEKUCE TESTU 7. 12](#_Toc170499538)

[7.3. BUG REPORT: 12](#_Toc170499539)

[8. TESTOVACÍ SCÉNÁŘ 13](#_Toc170499540)

[8.1. Test the DELETE method 13](#_Toc170499541)

[8.2. EXEKUCE TESTU 8. 13](#_Toc170499542)

[8.3. BUG REPORT: 13](#_Toc170499543)

# 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: xxxxx Username: xxxxxx Password: xxxxxxxxx |
| REST-API | <http://108.143.193.45:8080/api/v1/students/> |

## Poznámky:

Nezapomeňte, že v IT se data musí někde uložit a poté získat. Proto ověřte, že data jsou správně uložena a získávána z databáze.

Nezapomeňte do testovacích scénářů uvést testovací data, očekávaný výsledek včetně těla odpovědi a stavových kódů.

# 1. TESTOVACÍ SCÉNÁŘ

## 1.1. Test the Get method

*Abstract: Get data about existing student*

*Step 1.) GET* [*http://108.143.193.45:8080/api/v1/students/219*](http://108.143.193.45:8080/api/v1/students/219)

*Step 2.) Click on Send button*

*expected result:*

*1.) status code 200 ok*

*2.) Response*

*{*

*"id": 219,*

*"firstName": "",*

*"lastName": "JOSHUA",*

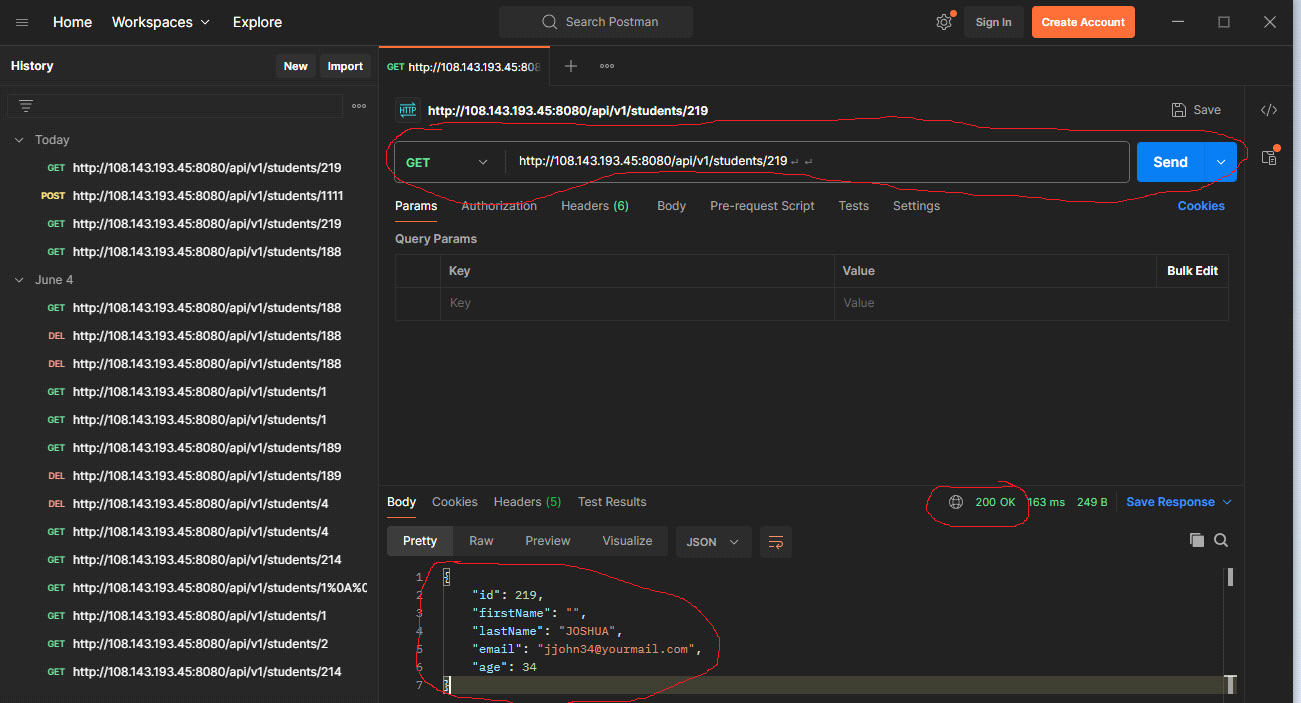
*"email": "jjohn34@yourmail.com",*

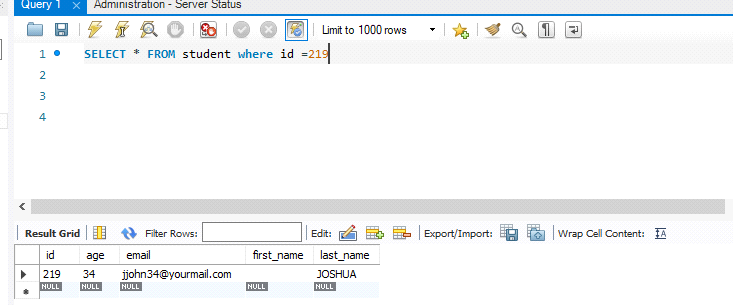
*"age": 34*

*}*

## 1.2. EXEKUCE TESTU 1.

*Result 1.test:*





## 1.3. BUG REPORT*:*

* No bug.

# 2. TESTOVACÍ SCÉNÁŘ

## 2.1. Test the Get method

Abstract: Get data about non-existing student

Step 1.) GET <http://108.143.193.45:8080/api/v1/students/1>

Step 2.) Click on Send button

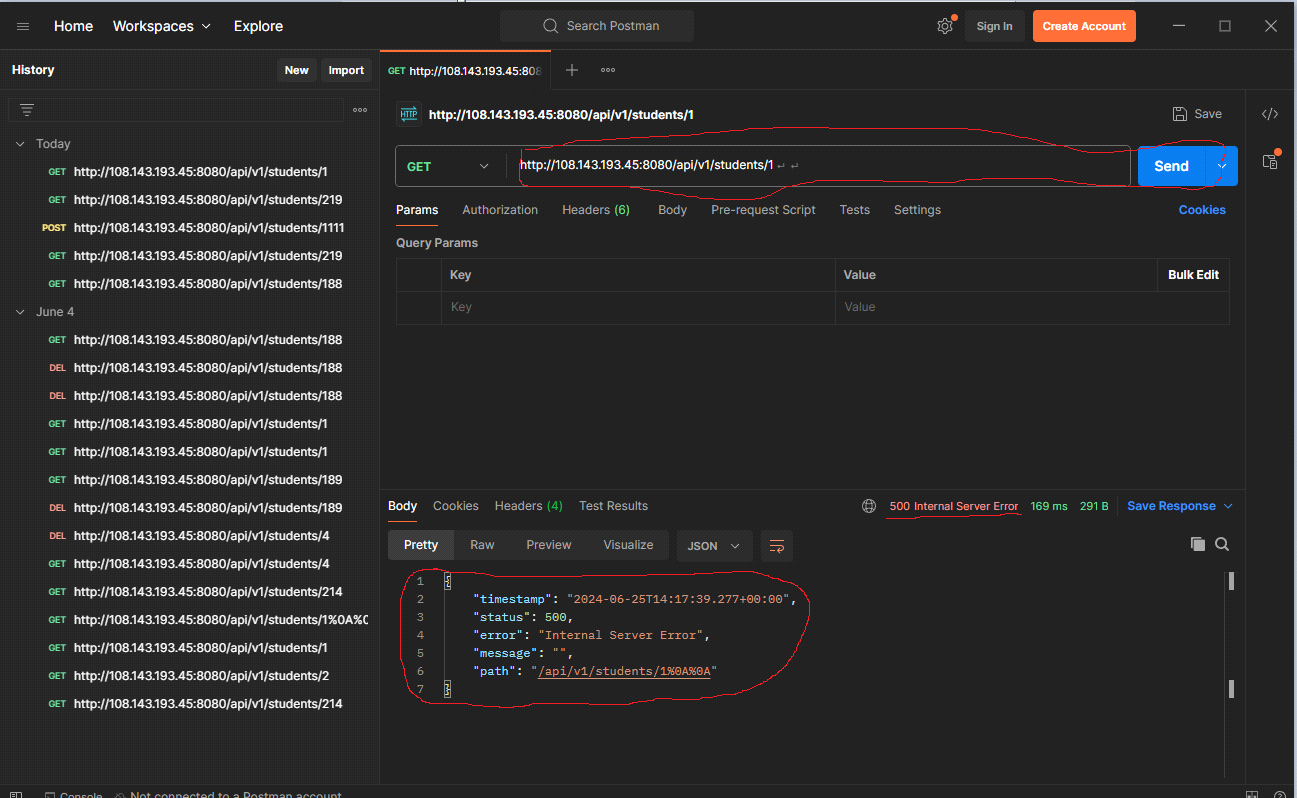
expected result:

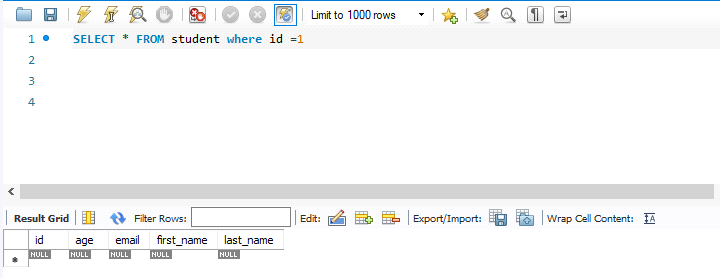
1.) status code 404 not found

2.) Response :

## 2.2. EXEKUCE TESTU 2.

*Result 2.test:*





## 3.2. BUG REPORT*:*

* Bug from 2. test: status code 500 internal Server Error

# 3. TESTOVACÍ SCÉNÁŘ

## 3.1. Test the POST method

Abstract: Create a new student

Step 1.) POST <http://108.143.193.45:8080/api/v1/students/>

Step 2.) Request body JSON

{

"firstName": "Pepa",

"lastName": "zdepa",

"email": "Pz@mail.com",

"age": 55

}

Step 3.) Click on Send button

expected result:

1.) status code 201 created

2.) Response :

{

"id": 1295,

"firstName": "pepa",

"lastName": "zdepa",

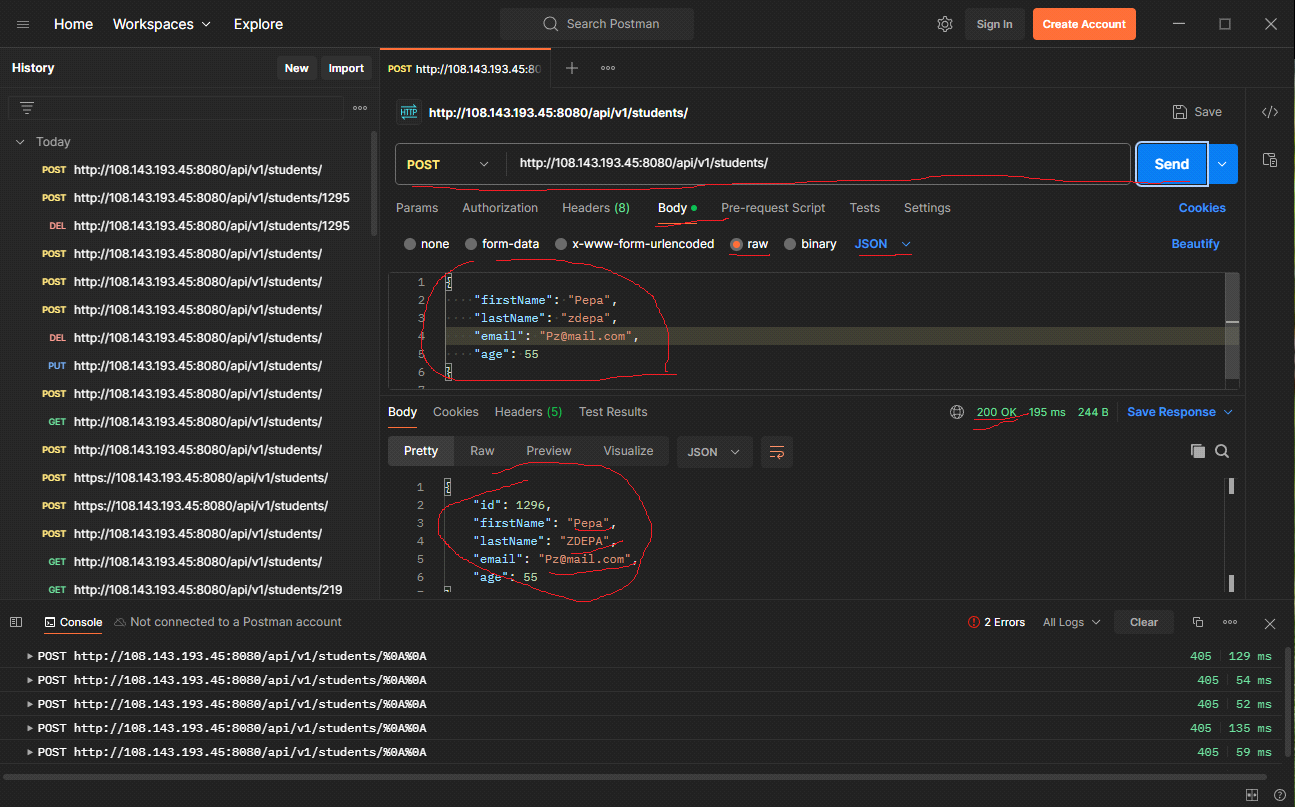
"email": "pz@mail.com",

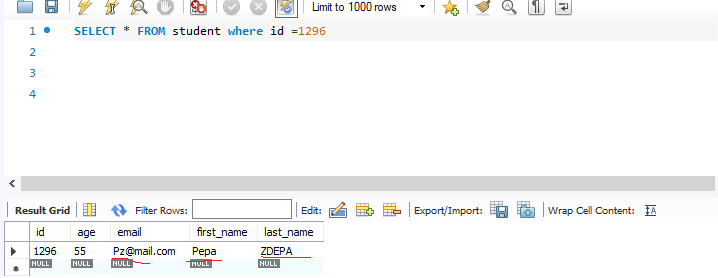
"age": 55

}

## 3.2. EXEKUCE TESTU 3.

*Result 3.test:*





## 3.3. BUG REPORT:

* Bug from 3. test: status code 200 ok
* Attributes firstName, lastName and email are not saved to the format lowerCase()

# 4. TESTOVACÍ SCÉNÁŘ

## 4.1. Test the POST method

Abstract: Creating a new student with a missing attribute

Step 1.) POST <http://108.143.193.45:8080/api/v1/students/>

Step 2.) Request body JSON

{

"firstName": "Lojza",

"lastName": "Koza",

"age": 5

}

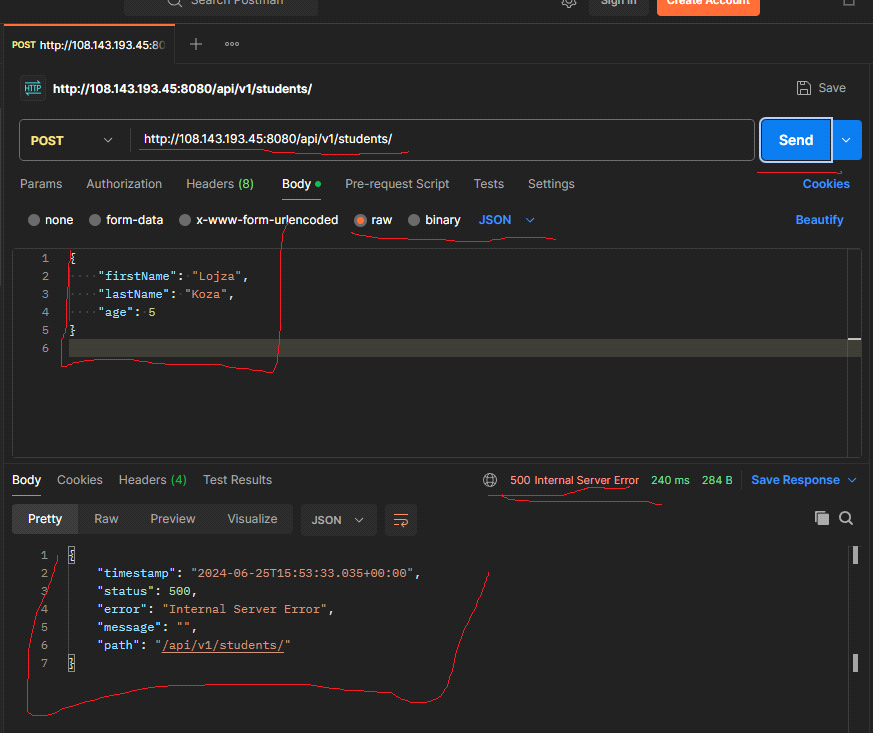
Step 3.) Click on Send button

expected result:

1.) status code 400 bad request (email are mandatory)

## 4.2. EXEKUCE TESTU 4.

*Result 4.test:*



## 4.3.3 BUG REPORT:

* Status code 500 Internal Server Error

# 5. TESTOVACÍ SCÉNÁŘ

## 5.1. Test the POST method

Abstract: Create a new student with a disallowed attribute.

Step 1.) POST <http://108.143.193.45:8080/api/v1/students/>

Step 2.) Request body JSON

{

"firstName": "hugo",

"lastName": "boss",

"email": "ssss",

"age": 55555

}

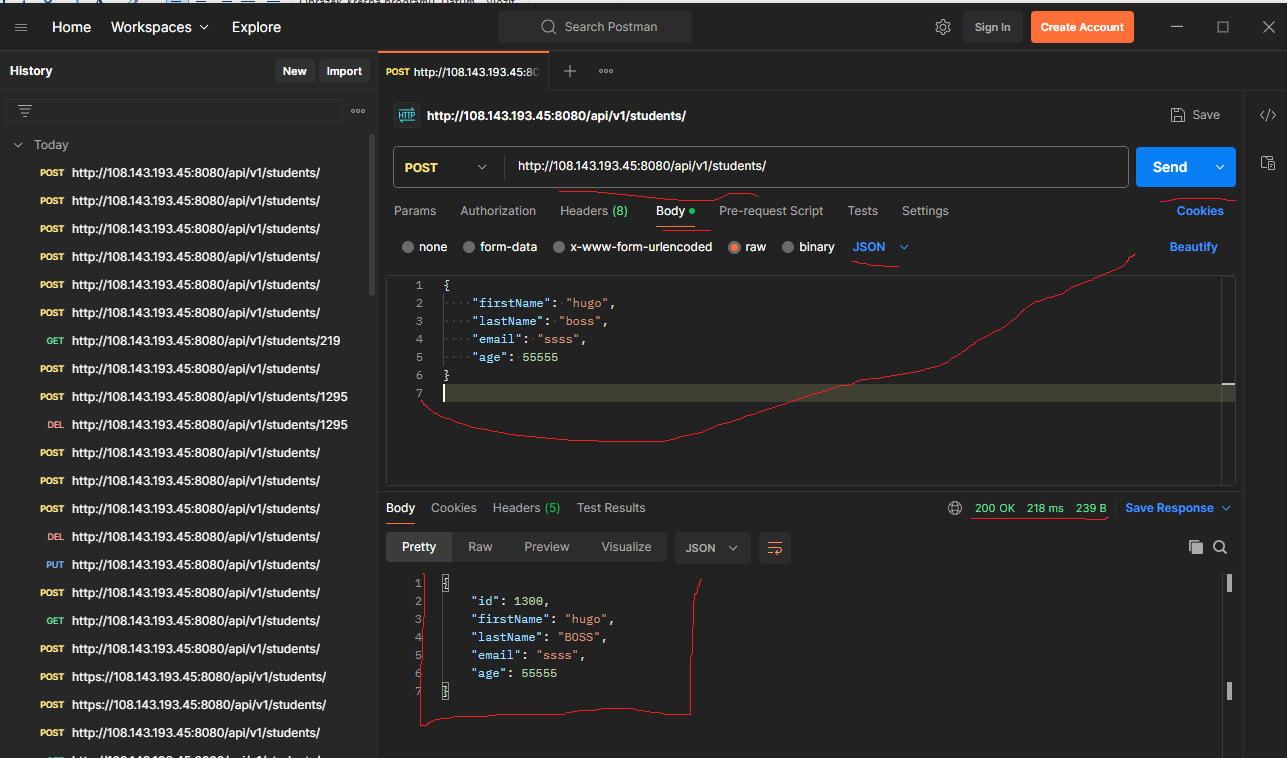
Step 3.) Click on Send button

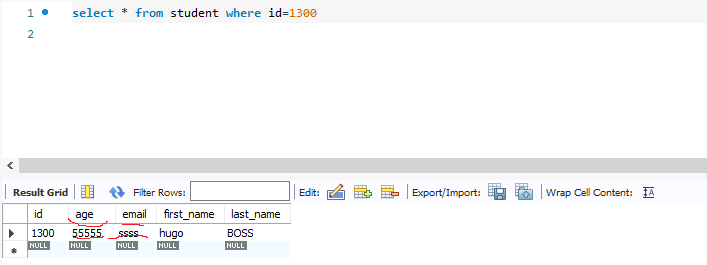
expected result:

1.) status code 400 bad request (Email and age is in the wrong format)

## 5.2. EXEKUCE TESTU 5.

*Result 5.test:*





## 5.3. BUG REPORT:

* Status code 200 ok
* Student was created.

# 6. TESTOVACÍ SCÉNÁŘ

## 6.1. Test the POST method

Abstract: Create a new student with a disallowed attribute.

Step 1.) POST <http://108.143.193.45:8080/api/v1/students/>

Step 2.) Request body JSON

{

"firstName": "ig",

"lastName": "IG",

"email": "ig@yourmail.com",

"age": 1

}

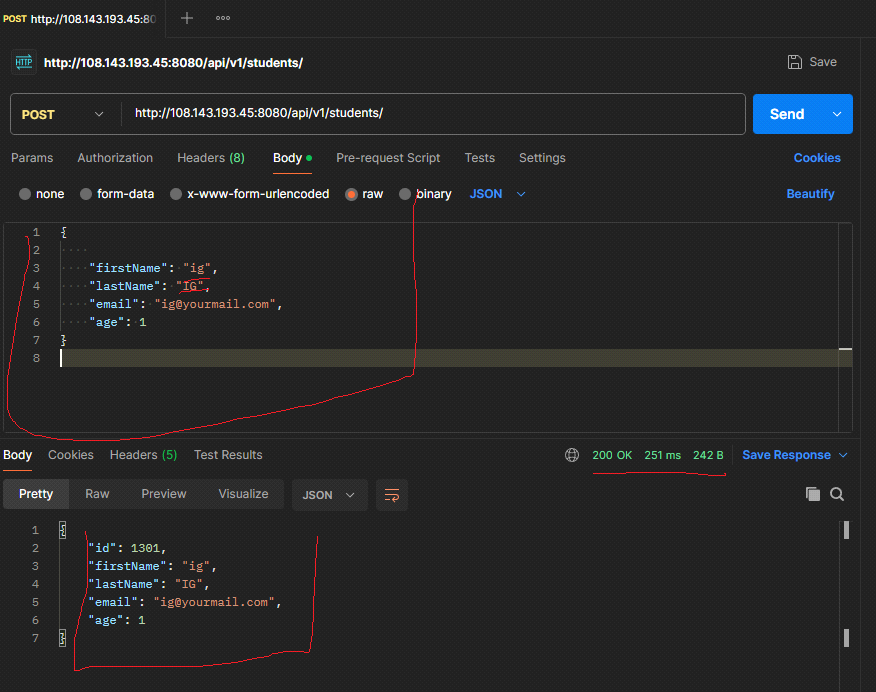
Step 3.) Click on Send button

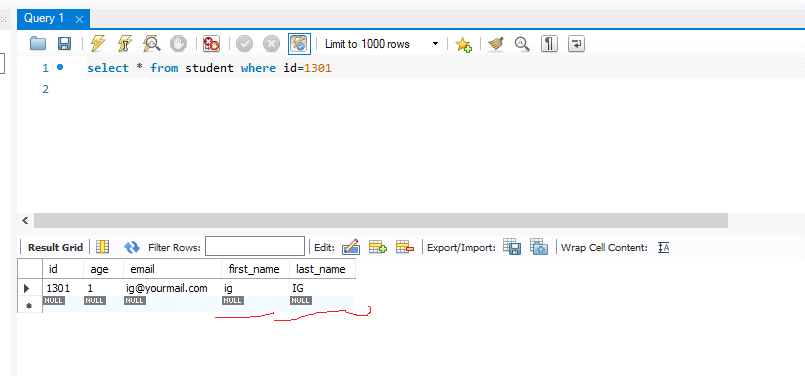
expected result:

1.) status code 400 bad request (firstName and lastName is in the wrong format)

## 6.2. EXEKUCE TESTU 6.

*Result 6.test:*





## 6.3. BUG REPORT:

* Status code 200 ok
* Student was created.

# 7. TESTOVACÍ SCÉNÁŘ

## 7.1 Test the DELETE method

Abstract: Delete existing student

Step 1.) DELETE <http://108.143.193.45:8080/api/v1/students/1301>

Step 2.) Click on Send button

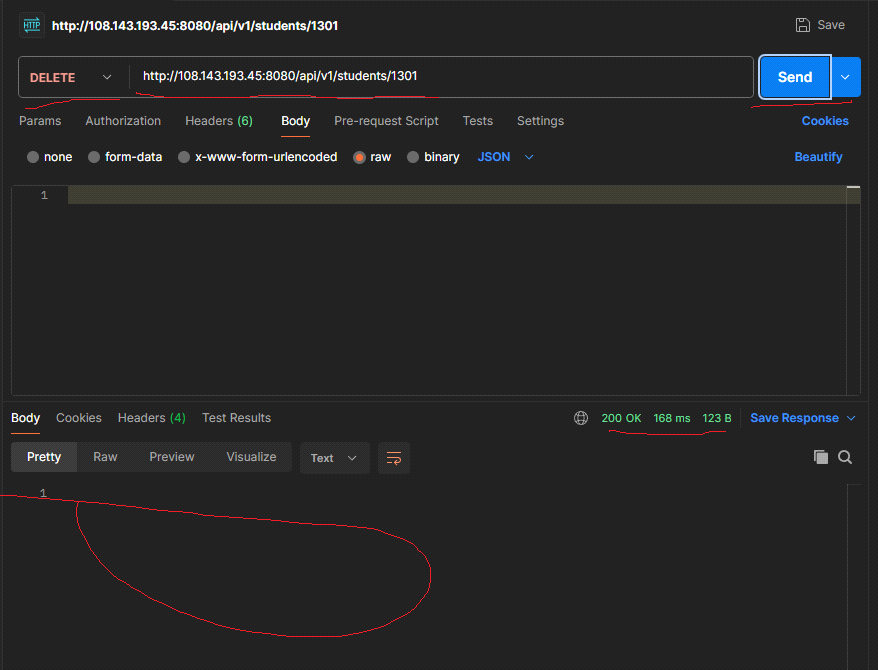
expected result:

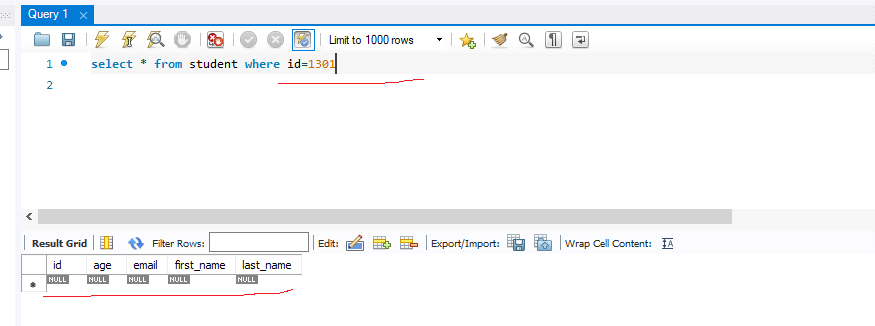
1.) status code 200

2.) All data about particular student data are REMOVED

## 7.2. EXEKUCE TESTU 7.

*Result 7.test:*





## 7.3. BUG REPORT:

* No bug.

# 8. TESTOVACÍ SCÉNÁŘ

## 8.1. Test the DELETE method

Abstract: Delete non-existing student

Step 1.) DELETE <http://108.143.193.45:8080/api/v1/students/1>

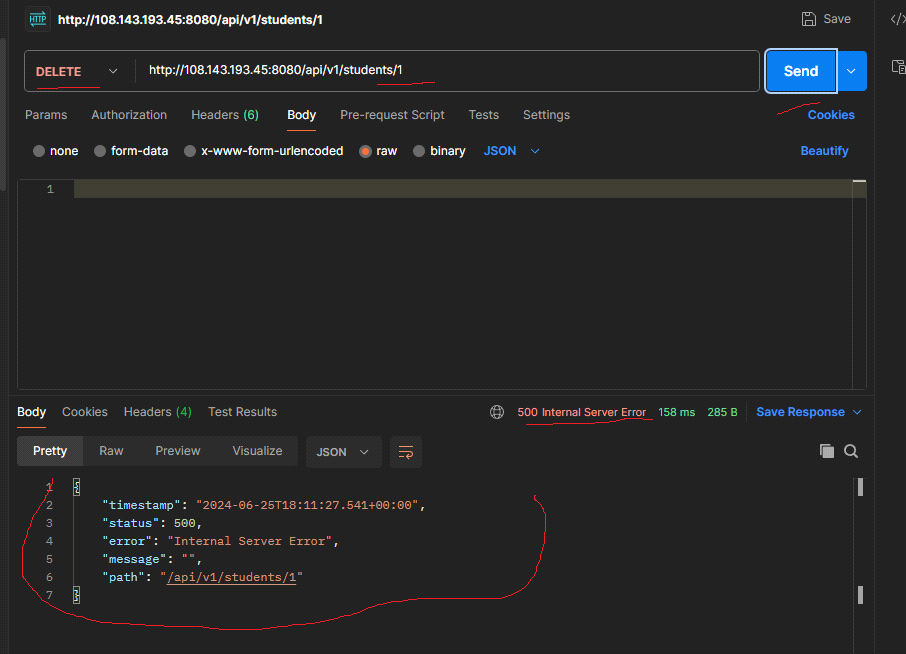
Step 2.) Click on Send button

expected result:

1.) status code 404 not found

## 8.2. EXEKUCE TESTU 8.

*Result 8.test:*



## 8.3. BUG REPORT:

* Status code 500 Internal Server Error