FINÁLNÍ PROJEKT   
 č.1



Autor: Veronika Pláničková/Hubková

Datum:28.10.2024

OBSAH

[**ZADÁNÍ**](#_heading=h.30j0zll) **3**

[**TESTOVACÍ SCÉNÁŘE**](#_heading=h.2et92p0) **4**

[**EXEKUCE TESTŮ**](#_heading=h.tyjcwt) **5**

[**BUG REPORT**](#_heading=h.3dy6vkm) **5**

# 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 | **Default scheme**: qa\_demo **Host**: aws.connect.psdb.cloud  **Port**: 3306 |
| 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.

:

# 

# TESTOVACÍ SCÉNÁŘE

*Na základě uvedených testovacích scénářů jsem ověřil(a) funkčnost aplikace.*

**GET**

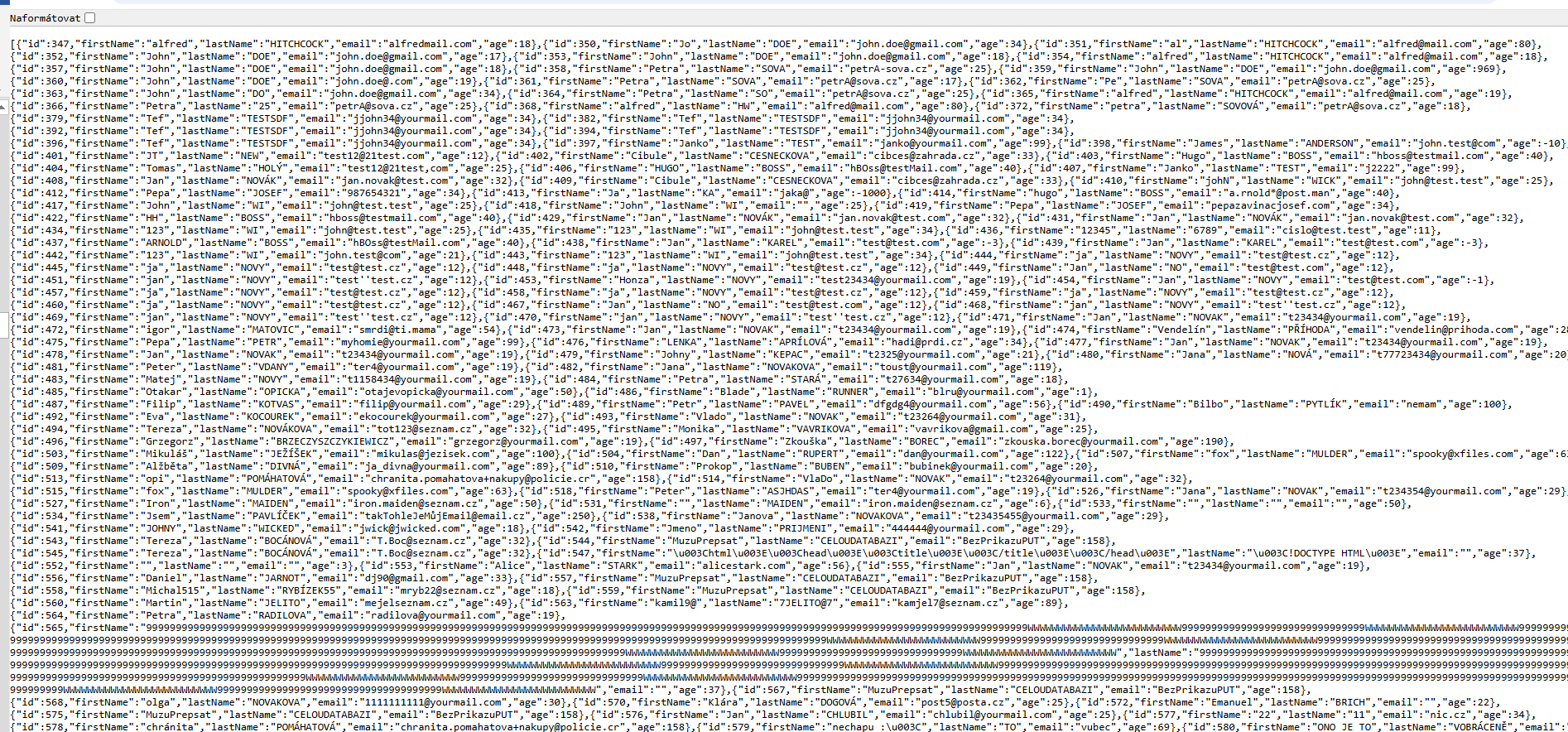
**(Postman)**

**Název:** Ověření získání celé databáze studentů.

**Test Case Number: 1.0**

**Kroky:**

1. Do webového prohlížeče Google Chrome zadám URL: <http://108.143.193.45:8080/api/v1/students/>.
2. Pro lepší přehlednost zvolím v nově otevřeném okně možnost „naformátovat“



1. V programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/> do prázdné kolonky a zvolím možnost „GET“.

**Očekávaný výsledek:**

Databáze ukáže všechny studenty, počáteční id 347 a poslední id 2012 (ke dni 28.10.2024)

    {

        "id": 347,

        "firstName": "alfred",

        "lastName": "HITCHCOCK",

        "email": "alfredmail.com",

        "age": 18

    },

    {

        "id": 2012,

        "firstName": "asdf",

        "lastName": "ASDF",

        "email": "asdf",

        "age": 35

    }

**(Dbeaver)**

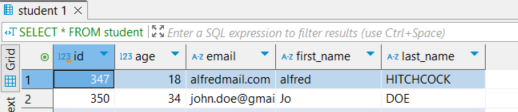
**Název:** Kontrola získání celé databáze studentů.

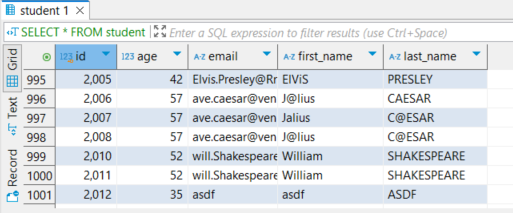
**Kroky:**

1. V programu Dbeaver vyhledám skrze příkaz SELECT \* FROM student

**Očekávaný výsledek:**

Zobrazení celé databáze studentů s dalšími parametry (id, age, email, first\_name, last\_name)





**GET**

**(Postman)**

**Název:** Ověření „získání“ správného studenta s id 347 z databáze studentů.

**Test Case Number: 1.1**

**Kroky:**

1. Do webového prohlížeče Google Chrome zadám URL: <http://108.143.193.45:8080/api/v1/students/>.
2. Pro lepší přehlednost zvolím v nově otevřeném okně možnost „naformátovat“
3. Zvolím si studenta, kterého chci v databázi vyhledat. Zvolení studenta s id 347.
4. V programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/347> do prázdné kolonky a zvolím možnost „GET“.

**Očekávaný výsledek:**

Databáze vyhledá na základě požadavku studenta s id 1957 a ten bude mít všechny vyplněné atributy (id, first\_name, last\_name, email, age)

    "id": 347,

    "firstName": "alfred",

    "lastName": "HITCHCOCK",

    "email": "alfredmail.com",

    "age": 18

**(Dbeaver)**

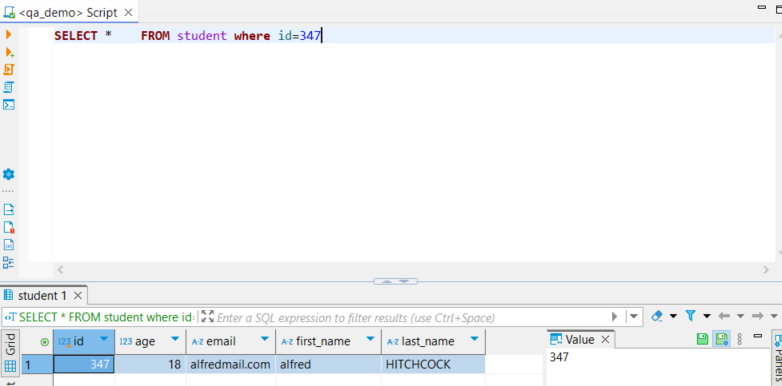
**Název:** Kontrola získání správného studenta s id 1957 v databázi studentů v programu Dbeaver.

**Kroky:**

1. V programu Dbeaver vyhledám skrze příkaz SELECT \* FROM student where id=347

**Očekávaný výsledek:**

Zobrazení správného studenta s id 347 s dalšími parametry (age, email, first\_name and last\_name).



**GET**

**(Postman)**

**Název:** Ověření „získání“ správného studenta s id 3000 z databáze studentů.

**Test Case Number: 1.2**

**Kroky:**

1. Do webového prohlížeče Google Chrome zadám URL: <http://108.143.193.45:8080/api/v1/students/>.
2. Pro lepší přehlednost zvolím v nově otevřeném okně možnost „naformátovat“
3. Zvolím si studenta, kterého chci v databázi vyhledat. Zvolení studenta s id 3000.
4. V programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/347> do prázdné kolonky a zvolím možnost „GET“.

**Očekávaný výsledek:**

Databáze ukáže chybovou hlášku „status“: 404, neboť takový student v databázi neexistuje.

{

    "timestamp": "2024-10-28T09:19:38.819+00:00",

    "status": 404,

    "error": "Internal Server Error",

    "message": "",

    "path": "/api/v1/students/3000"

}

**(Dbeaver)**

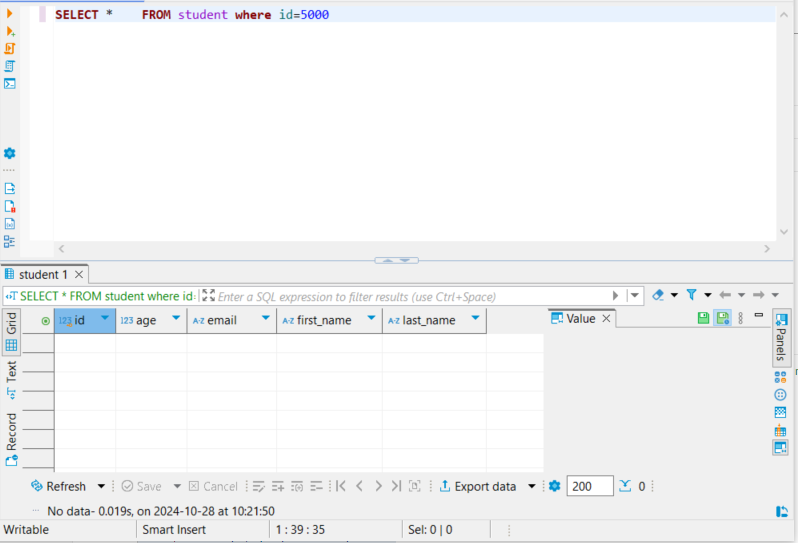
**Název:** Kontrola získání studenta s id 3000 v databázi studentů v programu Dbeaver.

**Kroky:**

1. V programu Dbeaver vyhledám skrze příkaz SELECT \* FROM student where id=3000

**Očekávaný výsledek:**

Databáze nevypíše žádného studenta. Takový student neexistuje.



**POST**

**(Postman)**

**Název:** Přidání dalšího studenta do databáze studentů

**Test Case Number: 2.0**

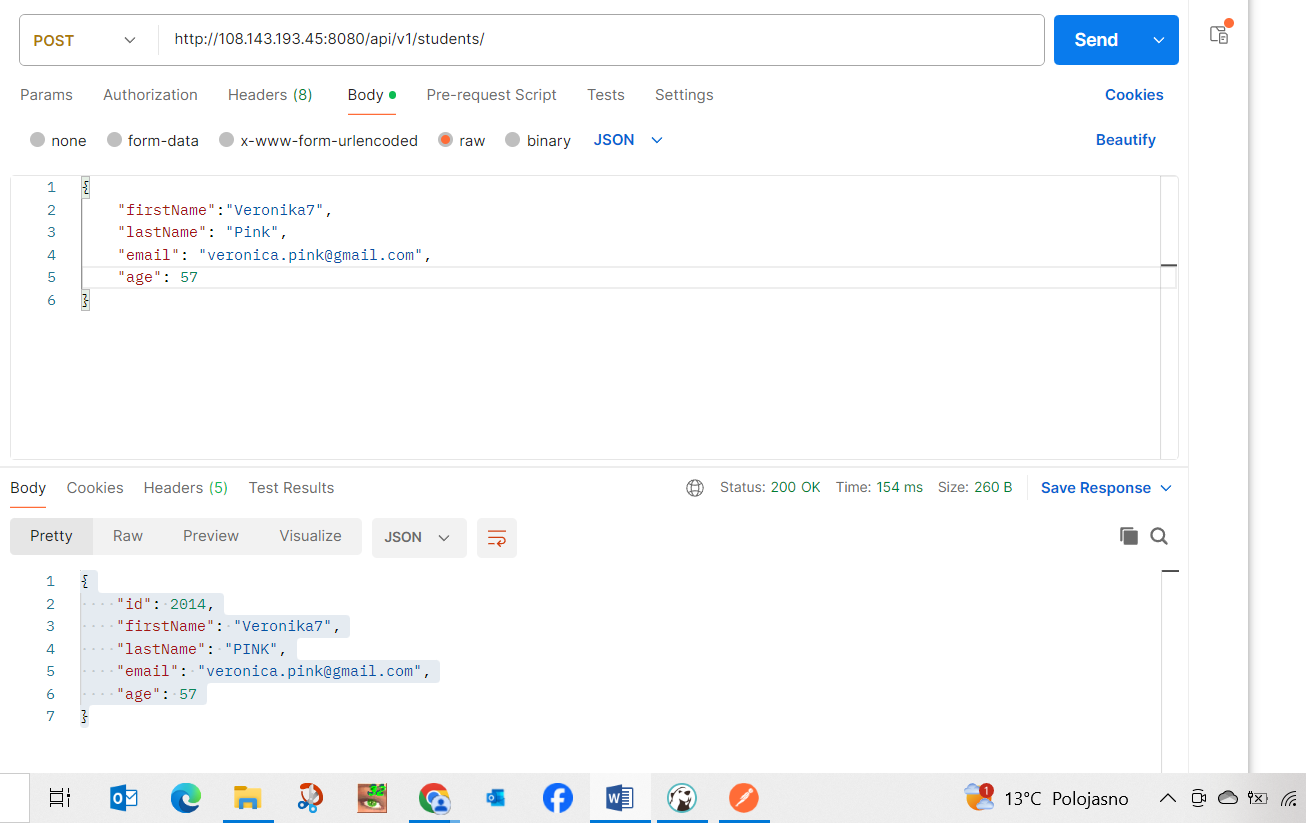
**Kroky:**

1. V  programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/> do prázdné kolonky a zvolím možnost „POST“.
2. Zvolím možnost „Body“ a typ „JSON“ a do těla pole napíši parametry s vyplněnými údaji:
3. {
4. "id": 2014,
5. "firstName": "Veronika7",
6. "lastName": "PINK",
7. "email": "veronica.pink@gmail.com",
8. "age": 57
9. }

3) Zvolím možnost „Send“.

**Očekávaný výsledek:**

Databáze bude obsahovat nového studenta s přiděleným id 2014 s dalšími vyplněnými parametry.



**(DBeaver)**

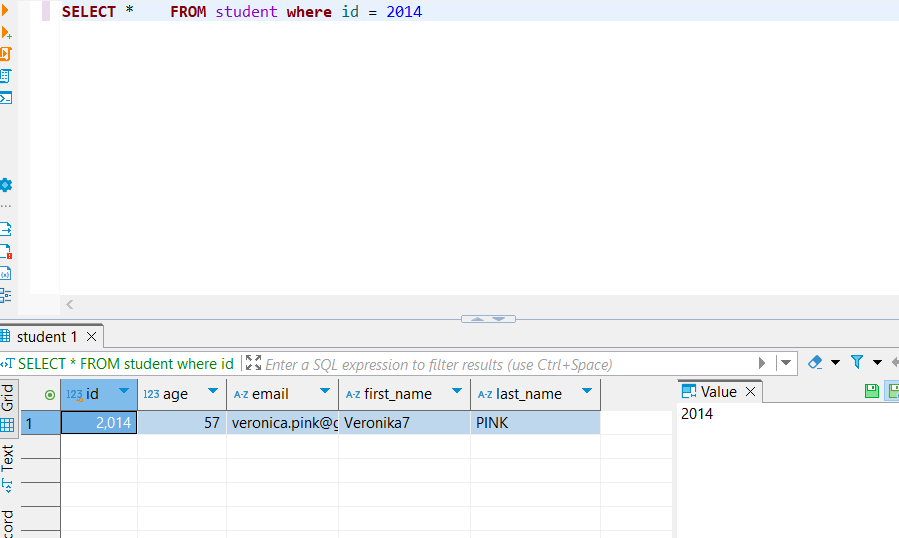
**Název:** Kontrola přidaného dalšího studenta do databáze studentů

**Kroky:**

1. V programu Dbeaver vyhledám pomocí příkazu SELECT \* FROM student where id=2014

**Očekávaný výsledek:**

Databáze bude obsahovat nového studenta s přiděleným id 2014 s dalšími vyplněnými parametry



**POST**

**(Postman)**

**Název:** Přidání dalšího studenta s jedním nevyplněným parametrem.

**Test Case Number: 2.1**

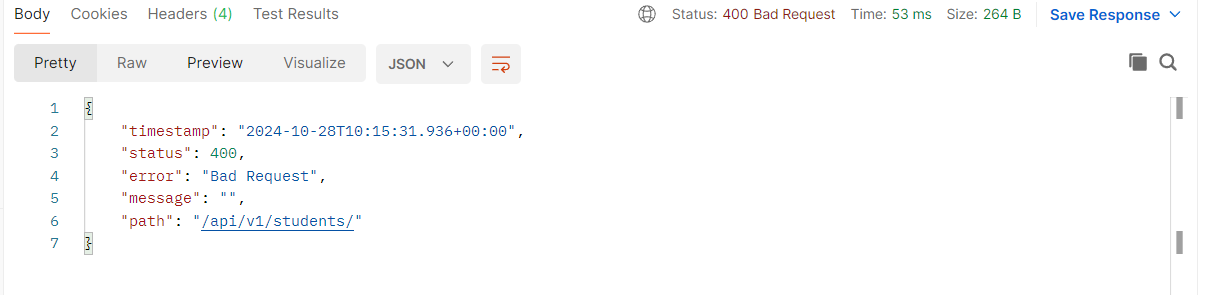
**Kroky:**

1. V  programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/> do prázdné kolonky a zvolím možnost „POST“.
2. Zvolím možnost „Body“ a typ „JSON“ a do těla pole vypíši parametry mimo jeden (age)
3. {
4. "firstName":"Veronika7",
5. "lastName": "Pink",
6. "email": "veronica.pink@gmail.com",
7. }

3) Zvolím možnost „Send“.

**Očekávaný výsledek:**

Naskočí chybová hláška s číslem 400, nesprávné zadání.



**DELETE**

**(Postman)**

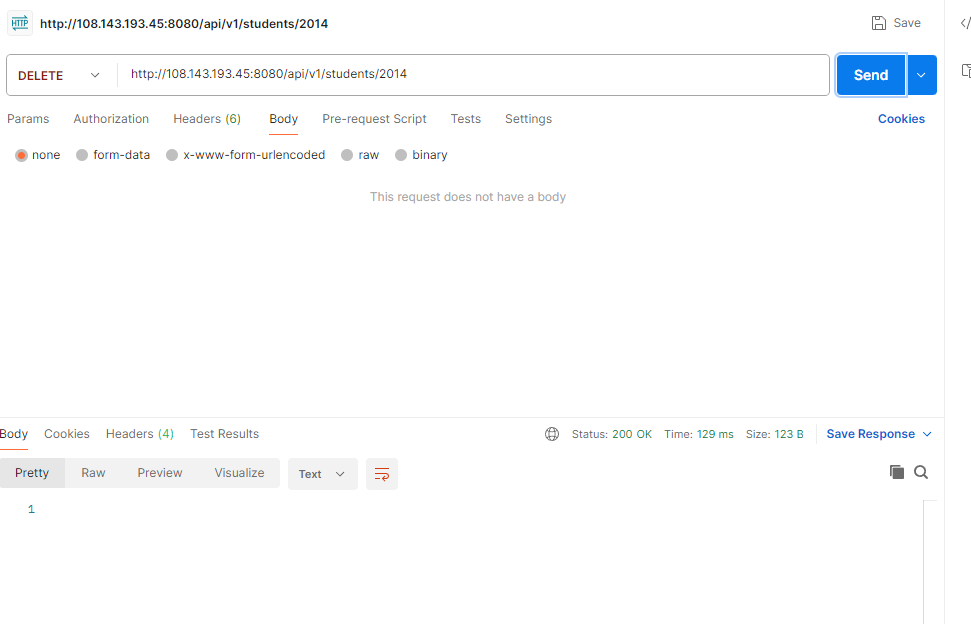
**Název:** Kontrola smazaného studenta s id 2014 v databázi studentů.

**Test Case Number: 3.0**

**Kroky:**

1. V Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/> s id 2014, které chci smazat.
2. Překliknu pole z „body“ na „none“
3. Zvolím možnost send.

**Očekáváný výsledek:** Student s id 2014 je smazán. Žádný student se neobjeví.



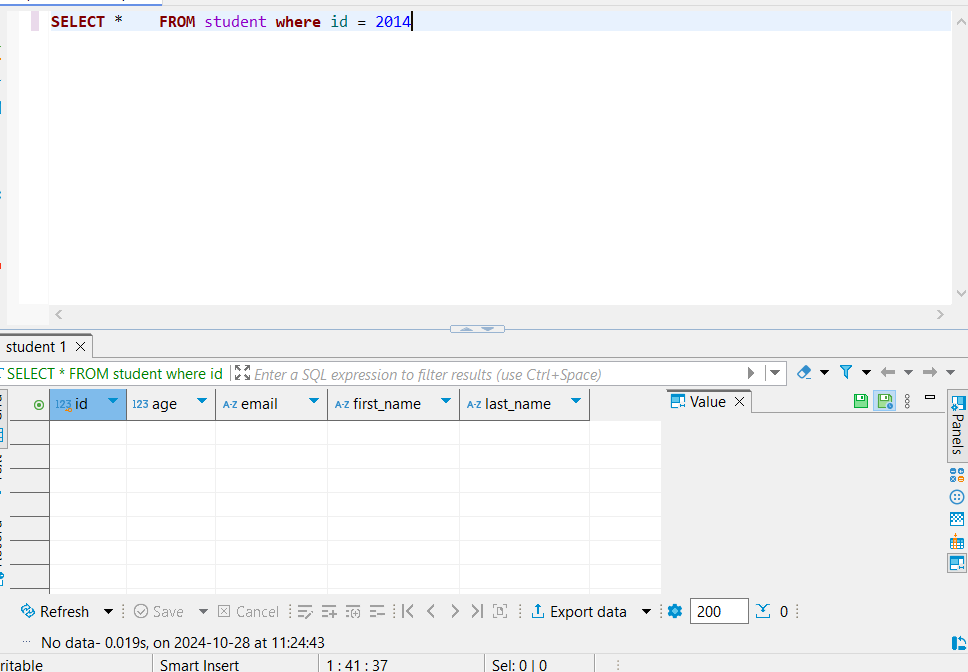
**(Dbeaver)**

**Název:** Kontrola smazaného studenta s id 2014 v databázi studentů.

**Kroky:**

1. V programu Dbeaver zadám příkaz: SELECT \* FROM student where id=2014

**Očekáváný výsledek:** Student s id 2014 je smazán. Nic se neobjeví.



EXEKUCE TESTŮ

*Testovací scénáře jsem provedl(a), přikládám výsledky testů.*

**GET**

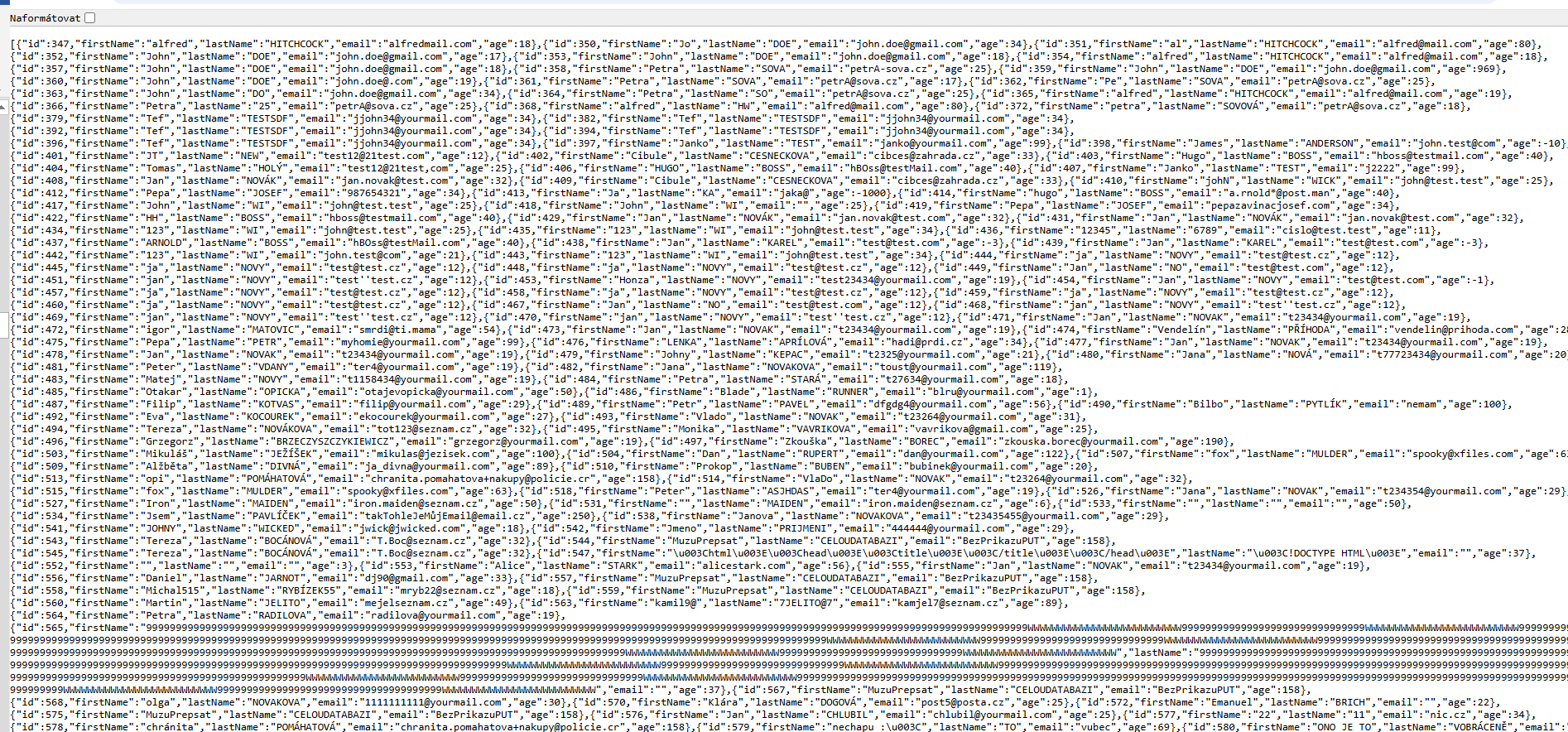
**(Postman)**

**Název:** Ověření získání celé databáze studentů.

**Test Case Number: 1.0**

**Kroky:**

1. Do webového prohlížeče Google Chrome zadám URL: <http://108.143.193.45:8080/api/v1/students/>.
2. Pro lepší přehlednost zvolím v nově otevřeném okně možnost „naformátovat“



1. V programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/> do prázdné kolonky a zvolím možnost „GET“.

**Očekávaný výsledek:**

Databáze ukáže všechny studenty, počáteční id 347 a poslední id 2012 (ke dni 28.10.2024)

    {

        "id": 347,

        "firstName": "alfred",

        "lastName": "HITCHCOCK",

        "email": "alfredmail.com",

        "age": 18

    },

    {

        "id": 2012,

        "firstName": "asdf",

        "lastName": "ASDF",

        "email": "asdf",

        "age": 35

    }

**Skutečný výsledek:**

Skutečný výsledek souhlasí s očekáváným výsledkem.

**(Dbeaver)**

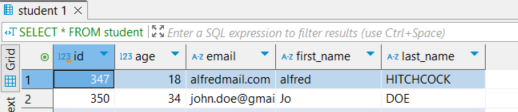
**Název:** Kontrola získání celé databáze studentů.

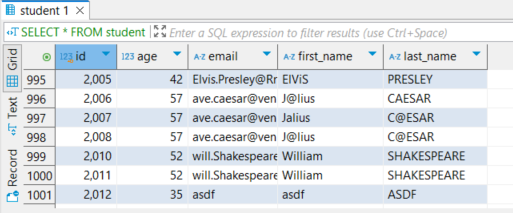
**Kroky:**

1. V programu Dbeaver vyhledám skrze příkaz SELECT \* FROM student

**Očekávaný výsledek:**

Zobrazení celé databáze studentů s dalšími parametry (id, age, email, first\_name, last\_name)





**Skutečný výsledek:**

Skutečný výsledek souhlasí s očekávaným výsledkem.

**GET**

**(Postman)**

**Název:** Ověření „získání“ správného studenta s id 347 z databáze studentů

**Test Case Number: 1.1**

**Kroky:**

1. Do webového prohlížeče Google Chrome zadám URL: <http://108.143.193.45:8080/api/v1/students/>.
2. Pro lepší přehlednost zvolím v nově otevřeném okně možnost „naformátovat“
3. Zvolím si studenta, kterého chci v databázi vyhledat. Zvolení studenta s id 347.
4. V programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/347> do prázdné kolonky a zvolím možnost „GET“.

**Očekávaný výsledek:**

Databáze vyhledá na základě požadavku studenta s id 1957 a ten bude mít všechny vyplněné atributy (id, first\_name, last\_name, email, age)

    "id": 347,

    "firstName": "alfred",

    "lastName": "HITCHCOCK",

    "email": "alfredmail.com",

    "age": 18

**Skutečný výsledek:**

Skutečný výsledek souhlasí s očekávaným výsledkem.

**(Dbeaver)**

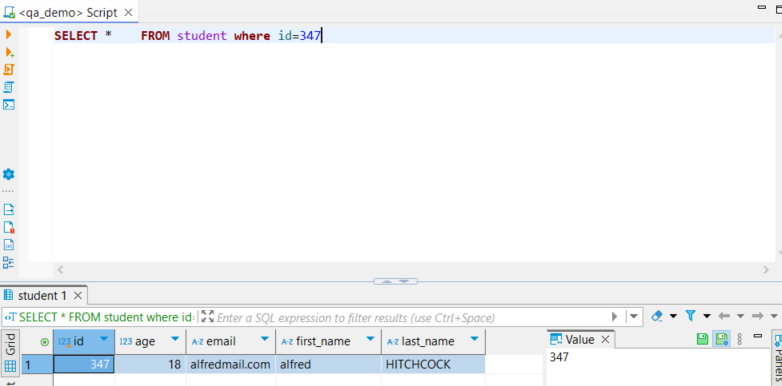
**Název:** Kontrola získání správného studenta s id 1957 v databázi studentů v programu Dbeaver.

**Kroky:**

1. V programu Dbeaver vyhledám skrze příkaz SELECT \* FROM student where id=347

**Očekávaný výsledek:**

Zobrazení správného studenta s id 347 s dalšími parametry (age, email, first\_name and last\_name).



**Skutečný výsledek:**

Skutečný výsledek souhlasí s očekávaným výsledkem.

**GET**

**(Postman)**

**Název:** Ověření „získání“ studenta s id 3000 z databáze studentů.

**Test Case Number: 1.2**

**Kroky:**

1. Do webového prohlížeče Google Chrome zadám URL: <http://108.143.193.45:8080/api/v1/students/>.
2. Pro lepší přehlednost zvolím v nově otevřeném okně možnost „naformátovat“
3. Zvolím si studenta, kterého chci v databázi vyhledat. Zvolení studenta s id 3000.
4. V programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/347> do prázdné kolonky a zvolím možnost „GET“.

**Očekávaný výsledek:**

Databáze ukáže chybovou hlášku „status“: 404, neboť takový student v databázi neexistuje.

{

    "timestamp": "2024-10-28T09:19:38.819+00:00",

    "status": 404,

    "error": "Internal Server Error",

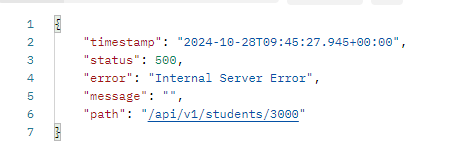
    "message": "",

    "path": "/api/v1/students/3000"

}

**Skutečný výsledek:**

Skutečný výsledek nesouhlasí s očekávaným výsledkem. Objevila se špatná chybová hláška s číslem 500. Nahlášeno v bug reportu.

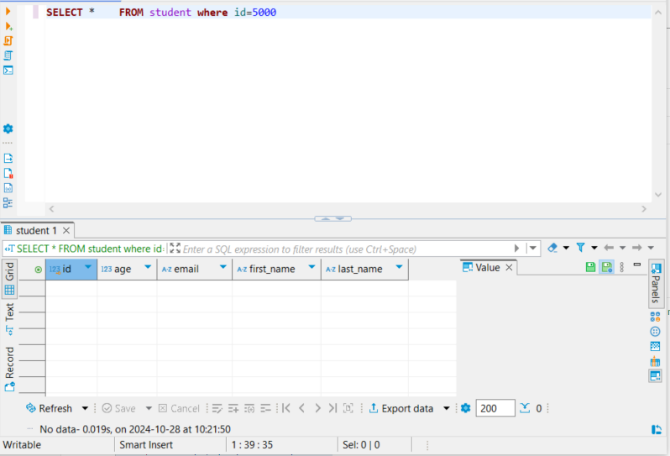


**(Dbeaver)**

**Název:** Kontrola získání studenta s id 3000 v databázi studentů v programu Dbeaver.

**Kroky:**

1. V programu Dbeaver vyhledám skrze příkaz SELECT \* FROM student where id=3000

**Očekávaný výsledek:**

Databáze nevypíše žádného studenta. Takový student neexistuje.

**Skutečný výsledek:**

Skutečný výsledek souhlasí s očekávaným výsledkem.

**POST**

**(Postman)**

**Název:** Přidání dalšího studenta do databáze studentů

**Test Case Number: 2.0**

**Kroky:**

1. V  programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/> do prázdné kolonky a zvolím možnost „POST“.
2. Zvolím možnost „Body“ a typ „JSON“ a do těla pole napíši parametry s vyplněnými údaji:
3. {
4. "id": 2014,
5. "firstName": "Veronika7",
6. "lastName": "PINK",
7. "email": "veronica.pink@gmail.com",
8. "age": 57
9. }

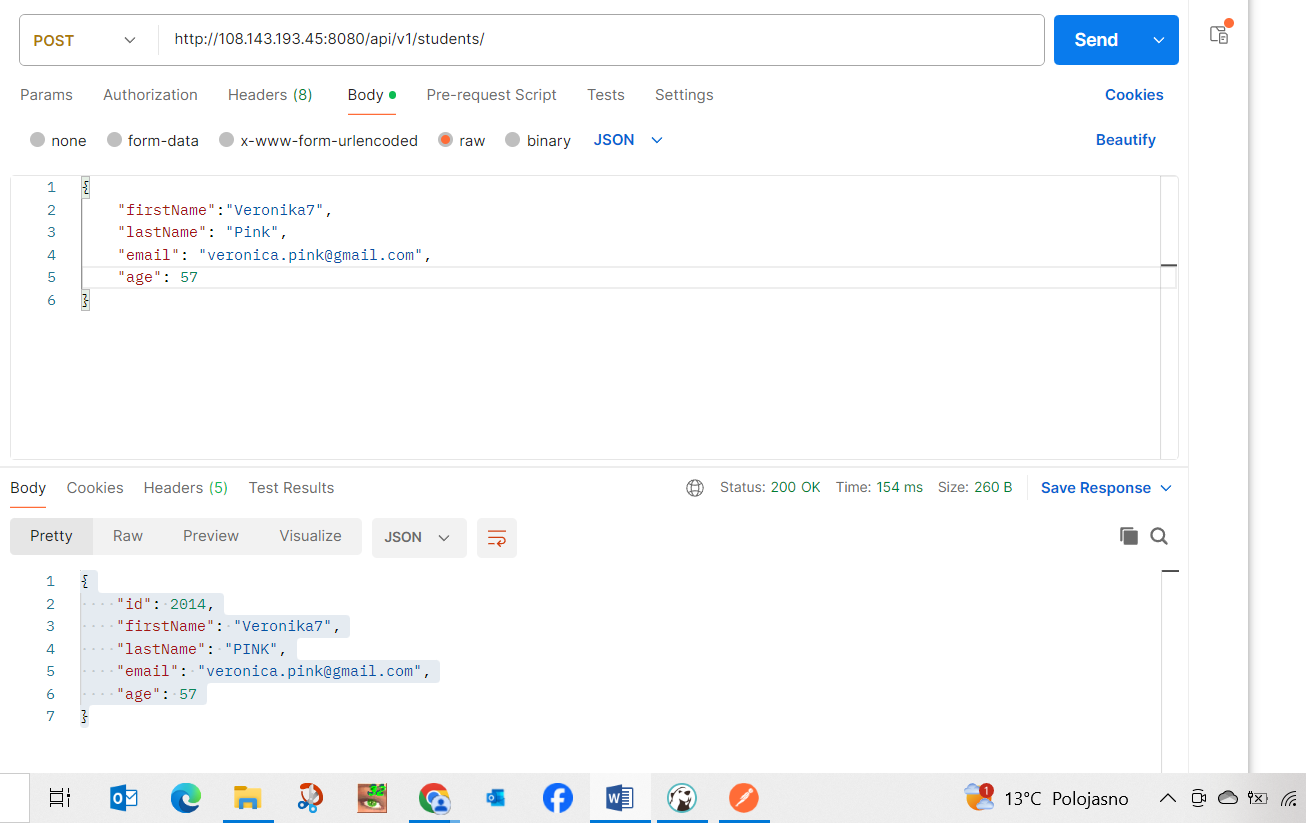
3) Zvolím možnost „Send“.

**Očekávaný výsledek:**

Databáze bude obsahovat nového studenta s přiděleným id 2014 s dalšími vyplněnými parametry.

**Skutečný výsledek:**

Skutečný výsledek souhlasí s očekávaným výsledkem.



**(DBeaver)**

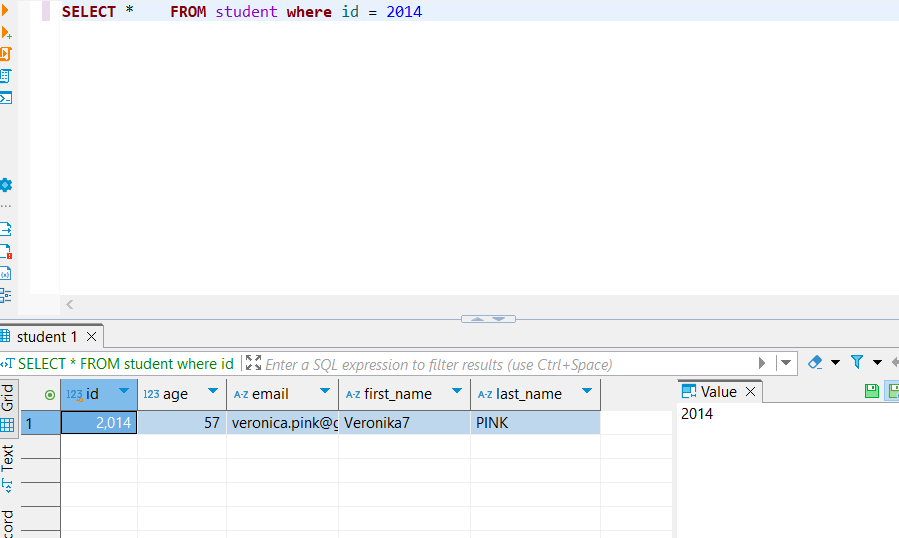
**Název:** Kontrola přidaného dalšího studenta do databáze studentů

**Kroky:**

1. V programu Dbeaver vyhledám pomocí příkazu SELECT \* FROM student where id=2014

**Očekávaný výsledek:**

Databáze bude obsahovat nového studenta s přiděleným id 2014 s dalšími vyplněnými parametry



**Skutečný výsledek:**

Skutečný výsledek souhlasí s očekávaným výsledkem.

**POST**

**(Postman)**

**Název:** Přidání dalšího studenta s jedním nevyplněným parametrem.

**Test Case Number: 2.1**

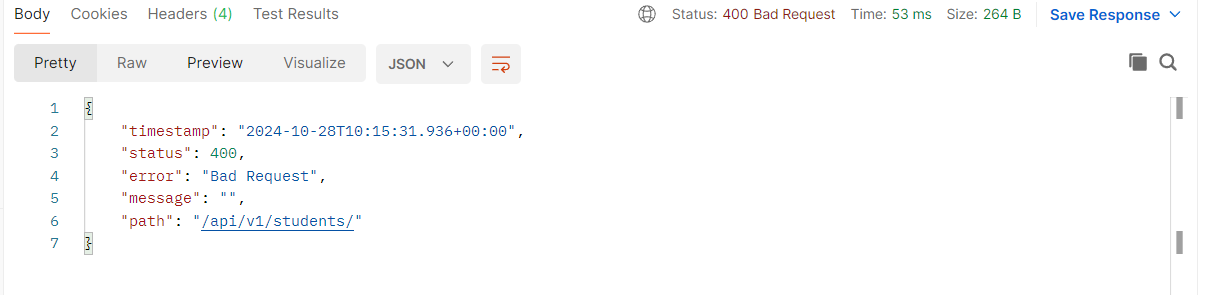
**Kroky:**

1. V  programu Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/> do prázdné kolonky a zvolím možnost „POST“.
2. Zvolím možnost „Body“ a typ „JSON“ a do těla pole vypíši parametry mimo jeden (age)
3. {
4. "firstName":"Veronika7",
5. "lastName": "Pink",
6. "email": "veronica.pink@gmail.com",
7. }

3) Zvolím možnost „Send“.

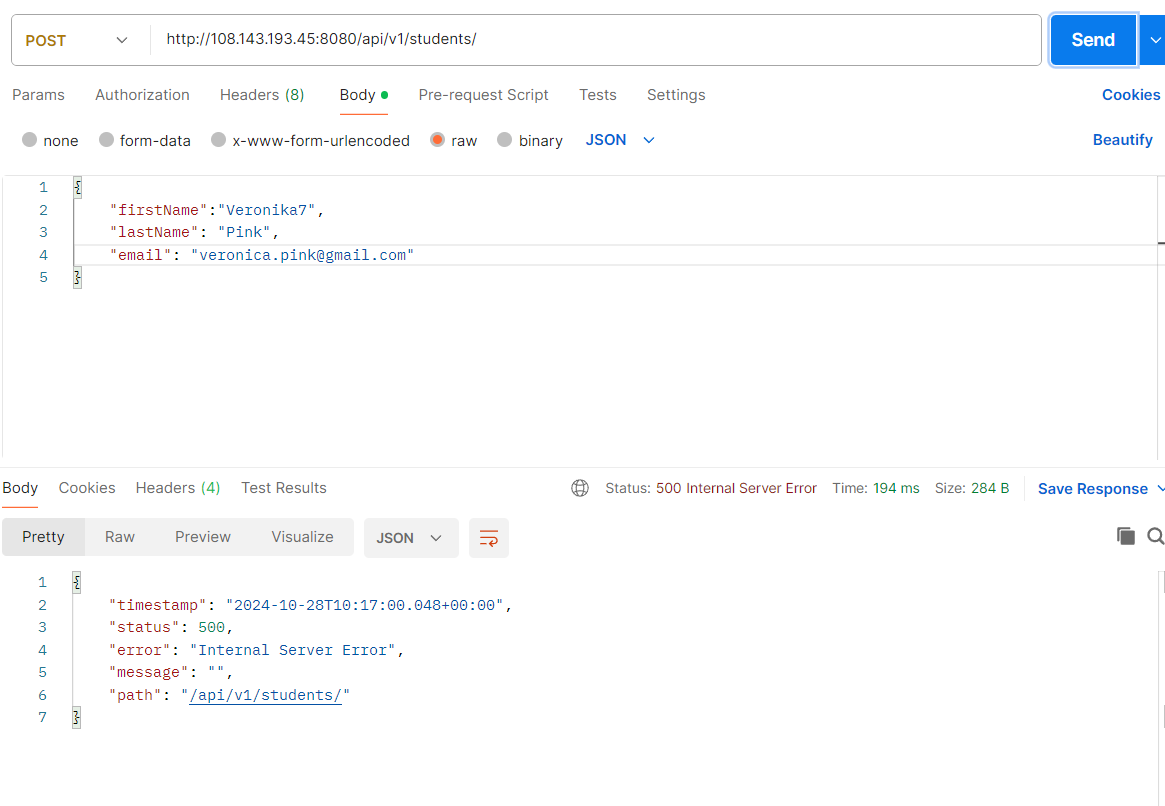
**Očekávaný výsledek:**

Naskočí chybová hláška s číslem 400, nesprávné zadání na straně uživatele.



**Skutečný výsledek:**

Skutečný výsledek se liší. Nesprávné zadání chybové hlášky, naskočilo „status“:500. Nahlášeno v bug reportu.



**DELETE**

**(Postman)**

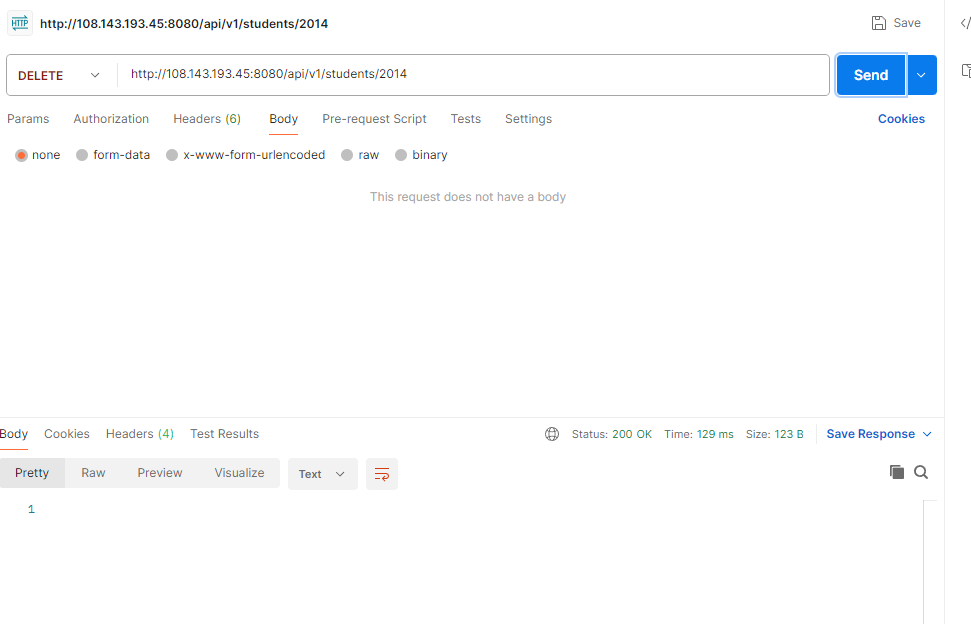
**Název:** Kontrola smazaného studenta s id 2014 v databázi studentů.

**Test Case Number: 3.0**

**Kroky:**

1. V Postman zadám URL adresu: <http://108.143.193.45:8080/api/v1/students/> s id 2014, které chci smazat.
2. Překliknu pole z „body“ na „none“
3. Zvolím možnost send.

**Očekáváný výsledek:** Student s id 2014 je smazán. Žádný student se neobjeví.



**Skutečný výsledek:**

Skutečný výsledek souhlasí s očekávaným výsledkem.

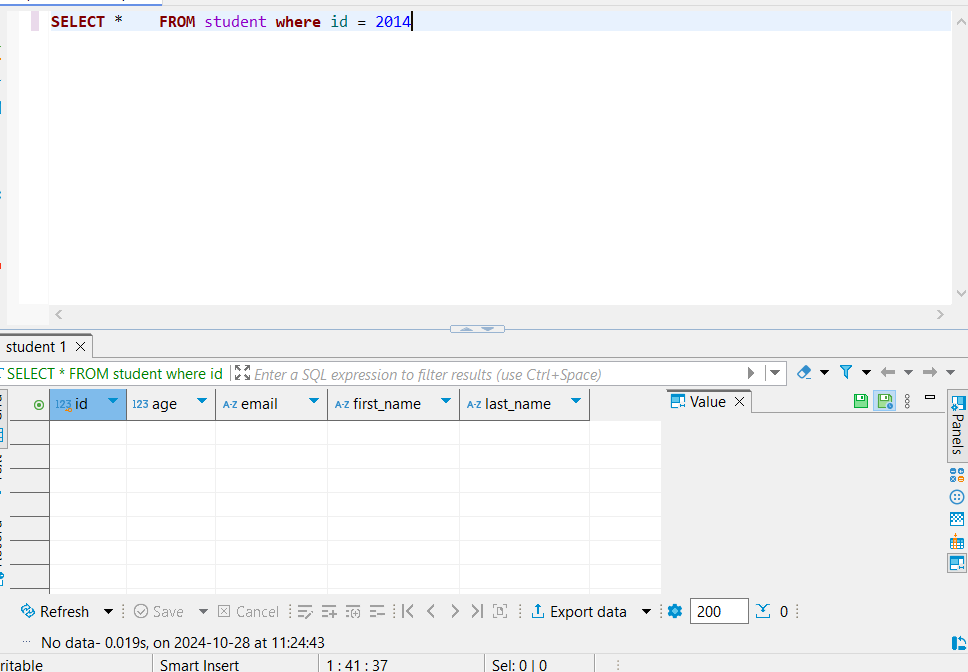
**(Dbeaver)**

**Název:** Kontrola smazaného studenta s id 2014 v databázi studentů.

**Kroky:**

1. V programu Dbeaver zadám příkaz: SELECT \* FROM student where id=2014

**Očekáváný výsledek:** Student s id 2014 je smazán.



**Skutečný výsledek:** Student s id 2014 je smazán.

# BUG REPORT

*Na základě provedených scénářů jsem objevil(a) uvedené chyby aplikace.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **BUG** | **PROBABILITY** | **SEVERITY** | **TEST CASE NUMBER** | **MITIGACE** |
| **GET**:  Špatná chybová hláška | vysoká | nízká | 1.2 | Opravení kodu |
| **POST:**  Špatná chybová hláška | vysoká | nízká | 2.1 | Opravení kodu |