FINÁLNÍ PROJEKT   
 č.1



Autor: Helga Vrbková  
Datum:

G

# Obsah

[ZADÁNÍ 4](#_Toc172283522)

[Přístupové údaje: 4](#_Toc172283523)

[Poznámky: 4](#_Toc172283524)

[TESTOVACÍ SCÉNÁŘE 5](#_Toc172283525)

[001 Abstract: GET data about existing student. 5](#_Toc172283526)

[002 Abstract: Get data about non-existing student 5](#_Toc172283527)

[003 Abstract: GET date of all students with „Last\_name“ = JOSHUA 6](#_Toc172283528)

[004 Abstract: DELETE all data about existing sudent 6](#_Toc172283529)

[005 DELETE all data about non-existing student 7](#_Toc172283530)

[006 DELETE all data of all students with the same „Last Name“ 7](#_Toc172283531)

[007 Abstract: Add a new student to the database 7](#_Toc172283532)

[008 Abstract: Create new student when all input data are the same as data of already existed student 8](#_Toc172283533)

[009 Abtract: Add a new student with some missing data 9](#_Toc172283534)

[010 Abstract: Add a new student with incorrect email address 9](#_Toc172283535)

[EXEKUCE TESTŮ 11](#_Toc172283536)

[001 Abstract: GET data about existing student. 11](#_Toc172283537)

[002 Abstract: Get data about non-existing student 12](#_Toc172283538)

[003 Abstract: GET date of all students with „Last\_name“ = JOSHUA 13](#_Toc172283539)

[004 Abstract: DELETE all data about existing sudent 14](#_Toc172283540)

[005 DELETE all data about non-existing student 15](#_Toc172283541)

[006 DELETE all data of all students with the same „Last Name“ 16](#_Toc172283542)

[007 Abstract: Add a new student to the database 17](#_Toc172283543)

[008 Abstract: Create new student when all input data are the same as data of already existed student 18](#_Toc172283544)

[009 Abtract: Add a new student with some missing data 20](#_Toc172283545)

[010 Abstract: Add a new student with incorrect email address 22](#_Toc172283546)

[BUG REPORT 24](#_Toc172283547)

[002 Abstract: Get data about nonexisting student 24](#_Toc172283548)

[003 Abstract: GET date of all students with „Last\_name“ = JOSHUA 25](#_Toc172283549)

[005 DELETE all data about non-existing student 26](#_Toc172283550)

[006 DELETE all data of all students with the same „Last Name“ 28](#_Toc172283551)

[007 Abstract: Add a new student to the database 29](#_Toc172283552)

[008 Abstract: Create new student when all input data are the same as data of already existed student 32](#_Toc172283553)

[009 Abtract: Add a new student with some missing data 34](#_Toc172283554)

[010 Abstract: Add a new student with incorrect email address 37](#_Toc172283555)

# 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/ |

## 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.*

|  |  |
| --- | --- |
| 001 Abstract: GET data about existing student. | |
| **Data Preparation:** | SELECT \* FROM student WHERE id=220;   |  |  |  |  |  | | --- | --- | --- | --- | --- | | id | age | Email | first\_name | last\_name | | 220 | 99 | [zkouska@yourmail.com](mailto:zkouska@yourmail.com) | John |  | |
| **Steps:** | GET: http://108.143.193.45:8080/api/v1/students/220 |
| **Expected result:** | 1. Status code 200 OK 2. Response   {  "id": 220,  "firstName": "John",  "lastName": "",  "email": "zkouska@yourmail.com",  "age": 99  } |

|  |  |
| --- | --- |
| 002 Abstract: Get data about non-existing student | |
| **Data Preparation:** | SELECT \* FROM student WHERE id=101;   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | id | age | email | first\_name | last\_name |  | | NULL | NULL | NULL | NULL | NULL |  | |
| **Steps:** | GET: http://108.143.193.45:8080/api/v1/students/101 |
| **Expected result:** | 1. Status code 404 Not Found 2. Response   {  "id": "",  "firstName": "",  "lastName": "",  "email": "",  "age": "",  } |

|  |  |
| --- | --- |
| 003 Abstract: GET date of all students with „Last\_name“ = JOSHUA | |
| **Data Preparation:** | SELECT \* FROM student WHERE last\_name = "JOSHUA";  A screenshot of a computer  Description automatically generated |
| **Steps:** | GET: http://108.143.193.45:8080/api/v1/students/JOSHUA |
| **Expected result:** | 1. Status code 200 OK 2. Response   Get 8 names with details: "id", "firstName", "lastName","email","age" for  "id": "219", "id": "228","id": "233","id": "264","id": "266","id": "268","id": "271","id": "1157". |

|  |  |
| --- | --- |
| 004 Abstract: DELETE all data about existing sudent | |
| **Data Preparation:** | SELECT \* FROM student WHERE id=238;   |  |  |  |  |  | | --- | --- | --- | --- | --- | | id | age | email | first\_name | last\_name | | 238 | 34 | m.novotna | Marie | NOVOTNÁ |   A screenshot of a computer  Description automatically generated |
| **Steps:** | DELETE: http://108.143.193.45:8080/api/v1/students/238 |
| **Expected result:** | 1. Status code 200 OK |

|  |  |
| --- | --- |
| 005 DELETE all data about non-existing student | |
| **Data Preparation:** | SELECT \* FROM student WHERE id=101;   |  |  |  |  |  | | --- | --- | --- | --- | --- | | id | age | email | first\_name | last\_name | | NULL | NULL | NULL | NULL | NULL | |
| **Steps:** | DELETE: http://108.143.193.45:8080/api/v1/students/101 |
| **Expected result:** | 1. Status code 404 Not Found |

|  |  |
| --- | --- |
| 006 DELETE all data of all students with the same „Last Name“ | |
| **Data Preparation:** | SELECT \* FROM student WHERE last\_name = “NEVĚŘÍCÍ“;  A screenshot of a computer  Description automatically generated |
| **Steps:** | DELETE: http://108.143.193.45:8080/api/v1/students/ last\_name = “NEVĚŘÍCÍ“; |
| **Expected result:** | 1. Status code 400 Bad Request 2. Response   Message: More than one student with the same LAST Name exists. Please use specific student ID. |

|  |  |
| --- | --- |
| 007 Abstract: Add a new student to the database | |
| **Data Preparation:** | POST – create new student Marek Benda, [marek.benda@seznam.cz](mailto:marek.benda@seznam.cz), 22 let  Conditions:   * all data are manatory (firstName, lastName, email, age) * all data are saved as lowercase * possible age 1 – 150 |
| **Steps:** | 1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON   {  "firstName": "Marek",  "lastName": "Benda",  "email": "[marek.benda@seznam.cz](mailto:marek.benda@seznam.cz)",  "age": "22"  } |
| **Expected result:** | 1. Status code 201created 2. Response   {  "id": xxxx,  "firstName": "Marek",  "lastName": "BENDA",  "email": "marek.benda@seznam.cz",  "age": 22  } |

|  |  |
| --- | --- |
| 008 Abstract: Create new student when all input data are the same as data of already existed student | |
| **Data Preparation:** | A screenshot of a computer  Description automatically generated  POST – create new student Marek Benda, [marek.benda@seznam.cz](mailto:marek.benda@seznam.cz), 22 let |
| **Steps:** | 1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON   {  "firstName": "Marek",  "lastName": "Benda",  "email": "[marek.benda@seznam.cz](mailto:marek.benda@seznam.cz)",  "age": "22"  } |
| **Expected result:** | 1. Status code 400 Bad request 2. Response   Message: The student already exists in the database. |

|  |  |
| --- | --- |
| 009 Abtract: Add a new student with some missing data | |
| **Data Preparation:** | POST – create new student: firstName missing, lastName Vopice, email missing, 78 let  Conditions:   * all data are manatory (firstName, lastName, email, age) * all data are saved as lowercase * possible age 1 – 150 |
| **Steps:** | **A)**  Steps:   1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON   {  "lastName": "Vopice",  "age": "78"  }  **B)**  Steps:   1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON   {  "firstName": "",  "lastName": "Vopice",  "email": "",  "age": "78"  } |
| **Expected result:** | Response:   1. Status 400 Bad Request |

|  |  |
| --- | --- |
| 010 Abstract: Add a new student with incorrect email address | |
| **Data Preparation:** | POST – create new student Václav Mrkvička, [Vaclav.mrkvicka.seznam.cz](mailto:marek.benda@seznam.cz), 33 let  Conditions:   * all data are manatory (firstName, lastName, email, age) * all data are saved as lowercase * possible age 1 – 150 |
| **Steps:** | 1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON   {  "firstName": "Václav",  "lastName": "Mrkvička",  "email": "[Vaclav.mrkvicka.seznam.cz](mailto:marek.benda@seznam.cz)",  "age": "33"  } |
| **Expected result:** | 1. Status code 400 Bad Request 2. Response   New student was not added to the database due to incorrect data: email address is not in proper form. |

# EXEKUCE TESTŮ

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

|  |
| --- |
| 001 Abstract: GET data about existing student. |
| Executed result:   1. GET: http://108.143.193.45:8080/api/v1/students/220 2. Press button SEND   Response:   1. Status code 200 OK 2. Response   {  "id": 220,  "firstName": "John",  "lastName": "",  "email": "zkouska@yourmail.com",  "age": 99  }  Test passed.  A screenshot of a computer  Description automatically generated |

|  |
| --- |
| 002 Abstract: Get data about non-existing student |
| Execuded result:   1. GET: http://108.143.193.45:8080/api/v1/students/101 2. Press button SEND   Response:   1. Status code 500 Internal Server Error 2. Response   {  "timestamp": "2024-06-17T12:05:34.956+00:00",  "status": 500,  "error": "Internal Server Error",  "message": "",  "path": "/api/v1/students/101"  }  Test faild.  A screenshot of a computer  Description automatically generated |

|  |
| --- |
| 003 Abstract: GET date of all students with „Last\_name“ = JOSHUA |
| Execuded result:   1. GET: http://108.143.193.45:8080/api/v1/students/JOSHUA 2. Press button SEND   Response:   1. Status code 400 Bad Request 2. Response   {  "timestamp": "2024-06-17T13:30:58.272+00:00",  "status": 400,  "error": "Bad Request",  "message": "",  "path": "/api/v1/students/JOSHUA"  }  Test faild.  A screenshot of a computer  Description automatically generated |

|  |
| --- |
| 004 Abstract: DELETE all data about existing sudent |
| Execuded result:   1. DELETE: http://108.143.193.45:8080/api/v1/students/238 2. Press button SEND   Response:   1. Status code 200 OK 2. Response   1   1. Checked again in My SQL Workbench 2. SELECT \* FROM student WHERE id=238; 3. Execude   A screenshot of a computer  Description automatically generated  Test passed. |

|  |
| --- |
| 005 DELETE all data about non-existing student |
| Execuded result:   1. DELETE: http://108.143.193.45:8080/api/v1/students/101 2. Press button SEND   Response:   1. Status code 500 Internal Server Error 2. Response   {  "timestamp": "2024-06-21T09:35:49.293+00:00",  "status": 500,  "error": "Internal Server Error",  "message": "",  "path": "/api/v1/students/101"  }A screenshot of a computer  Description automatically generated  Test faild. |

|  |
| --- |
| 006 DELETE all data of all students with the same „Last Name“ |
| Execuded result:   1. DELETE: http://108.143.193.45:8080/api/v1/students/ last\_name = “NEVĚŘÍCÍ“ 2. Press button SEND   Response:   1. Status code 400 Bad Request 2. Response   {  "timestamp": "2024-06-21T09:53:50.880+00:00",  "status": 400,  "error": "Bad Request",  "message": "",  "path": "/api/v1/students/last\_name%20=%20%22NEV%C4%9A%C5%98%C3%8DC%C3%8D%22;"  }A screenshot of a computer  Description automatically generated  Test faild. |

|  |
| --- |
| 007 Abstract: Add a new student to the database |
| Executed result:   1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON 3. {   "firstName": "Marek",  "lastName": "Benda",  "email": "marek.benda@seznam.cz",  "age": 22  }   1. SEND   Response:   1. Status 200 OK 2. {   "id": 1278,  "firstName": "Marek",  "lastName": "BENDA",  "email": "marek.benda@seznam.cz",  "age": 22  }  A screenshot of a computer  Description automatically generated   1. Database check   A screenshot of a computer  Description automatically generated  Test failed. |

|  |
| --- |
| 008 Abstract: Create new student when all input data are the same as data of already existed student |
| Executed result:   1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON 3. {   "firstName": "Marek",  "lastName": "Benda",  "email": "marek.benda@seznam.cz",  "age": 22  }   1. SEND   Response:  1) Status 200 OK  2) Response  {  "id": 1287,  "firstName": "Marek",  "lastName": "BENDA",  "email": "marek.benda@seznam.cz",  "age": 22  }  A screenshot of a computer  Description automatically generated3) Check database  A screenshot of a computer  Description automatically generated  Test faild. |

|  |
| --- |
| 009 Abtract: Add a new student with some missing data |
| **A)**  Executed steps:   1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON   {  "lastName": "Vopice",  "age": "78"  }   1. Press button SEND   Response:   1. Status 500 Internal Server Error 2. {   "timestamp": "2024-06-25T10:05:43.408+00:00",  "status": 500,  "error": "Internal Server Error",  "message": "",  "path": "/api/v1/students/"  }  A screenshot of a computer  Description automatically generated   1. Check database – student was not added to the database.   Test failed. |
| **B)**  Executed steps:   1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON   {  "firstName": "",  "lastName": "Vopice",  "email": "",  "age": "78"  }   1. Press button SEND   Response:   1. Status code 200 OK 2. Response:   {  "id": 1292,  "firstName": "",  "lastName": "VOPICE",  "email": "",  "age": 78  }  **A screenshot of a computer  Description automatically generated**   1. Check database   **A screenshot of a computer  Description automatically generated**  Test failed. |

|  |
| --- |
| 010 Abstract: Add a new student with incorrect email address |
| Executed Steps:   1. POST <http://108.143.193.45:8080/api/v1/students/> 2. Body + raw + JSON   {  "firstName": "Václav",  "lastName": "Mrkvička",  "email": "[Vaclav.mrkvicka.seznam.cz](mailto:marek.benda@seznam.cz)",  "age": "33"  }   1. Press button SEND   Response:   1. Status 200 OK   {  "id": 1293,  "firstName": "Václav",  "lastName": "MRKVIČKA",  "email": "Vaclav.mrkvicka.seznam.cz",  "age": 33  }  *A screenshot of a computer  Description automatically generated*   1. Checked database   *A screenshot of a computer  Description automatically generated*  Test failed. |

# BUG REPORT

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

## 002 Abstract: Get data about nonexisting student

Data Preparation:

SELECT \* FROM student WHERE id=101;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | age | email | first\_name | last\_name |
| NULL | NULL | NULL | NULL | NULL |

Steps:

GET: http://108.143.193.45:8080/api/v1/students/101

Reproduction steps:

1. GET: http://108.143.193.45:8080/api/v1/students/101
2. Press button SEND

Actual result:

1. Status code 500 Internal Server Error
2. Response

{

"timestamp": "2024-06-17T12:05:34.956+00:00",

"status": 500,

"error": "Internal Server Error",

"message": "",

"path": "/api/v1/students/101"

}

Test faild.

A screenshot of a computer

Description automatically generated

Expected result:

1. Status code 404 Not Found
2. Response

{

"timestamp": "2024-06-17T12:05:34.956+00:00",

"status": 404,

"error": "Not Found ",

"message": "Non-existing student",

"path": "/api/v1/students/101"

}

## 003 Abstract: GET date of all students with „Last\_name“ = JOSHUA

Data Preparation:

SELECT \* FROM student WHERE last\_name = "JOSHUA";

A screenshot of a computer

Description automatically generated

Steps:

GET: http://108.143.193.45:8080/api/v1/students/JOSHUA

Reproduction steps:

1. GET: http://108.143.193.45:8080/api/v1/students/JOSHUA
2. Press button SEND

Actual Result:

1. Status code 400 Bad Request
2. Response

{

"timestamp": "2024-06-17T13:30:58.272+00:00",

"status": 400,

"error": "Bad Request",

"message": "",

"path": "/api/v1/students/JOSHUA"

}

Test faild. In case the database contains more then one sutent with the same name (JOSHUA) the actual result is faild.

A screenshot of a computer

Description automatically generated

Expected result:

In case the database contains more then one student with the same name (JOSHUA) the result should contain all names and related "id", "firstName", "lastName","email","age".

1. Status code 200 OK
2. Response

Get 8 names with details: "id", "firstName", "lastName","email","age" for

"id": "219", "id": "228","id": "233","id": "264","id": "266","id": "268","id": "271","id": "1157".

Example of expected result:

{

"id": "219",

"firstName": "",

"lastName": "JOSHUA",

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

"age": "34",

}

## 005 DELETE all data about non-existing student

Data Preparation:

SELECT \* FROM student WHERE id=101;

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id | age | email | first\_name | last\_name |
| NULL | NULL | NULL | NULL | NULL |

Steps:

DELETE: http://108.143.193.45:8080/api/v1/students/101

Expected result:

1. Status code 404 Not Found

Reproduction steps:

1. DELETE: http://108.143.193.45:8080/api/v1/students/101
2. Press button SEND

Actual Result:

1. Status code 500 Internal Server Error
2. Response

{

"timestamp": "2024-06-21T09:35:49.293+00:00",

"status": 500,

"error": "Internal Server Error",

"message": "",

"path": "/api/v1/students/101"

}A screenshot of a computer

Description automatically generated

Test faild.

Expected Result:

1. Status code 404 Not Found
2. Response

{

"timestamp": "2024-06-21T09:35:49.293+00:00",

"status": 404,

"error": "Not Found",

"message": "Non-existing student",

"path": "/api/v1/students/101"

}

## 006 DELETE all data of all students with the same „Last Name“

Data Preparation:

SELECT \* FROM student WHERE last\_name = “NEVĚŘÍCÍ“;

A screenshot of a computer

Description automatically generated

Steps:

DELETE: http://108.143.193.45:8080/api/v1/students/ last\_name = “NEVĚŘÍCÍ“;

Expected result:

1. Status code 400 Bad Request
2. Response

Message: More than one student with the same LAST Name exists. Please use specific student ID.

Reproduction steps:

1. DELETE: http://108.143.193.45:8080/api/v1/students/ last\_name = “NEVĚŘÍCÍ“
2. Press button SEND

Actual Result:

1. Status code 400 Bad Request
2. Response

{

"timestamp": "2024-06-21T09:53:50.880+00:00",

"status": 400,

"error": "Bad Request",

"message": "",

"path": "/api/v1/students/last\_name%20=%20%22NEV%C4%9A%C5%98%C3%8DC%C3%8D%22;"

}

A screenshot of a computer

Description automatically generated

Test faild.

Expected result:

1. Status code 400 Bad Request
2. Response

{

"timestamp": "2024-06-21T09:53:50.880+00:00",

"status": 400,

"error": "Bad Request",

"message": "To be able to delete data of a student it is necessary to use related sudent’s ID",

"path": "/api/v1/students/last\_name%20=%20%22NEV%C4%9A%C5%98%C3%8DC%C3%8D%22;"

}

## 007 Abstract: Add a new student to the database

Data preparation:

POST – create new student Marek Benda, [marek.benda@seznam.cz](mailto:marek.benda@seznam.cz), 22 let

Conditions:

* all data are manatory (firstName, lastName, email, age)
* all data are saved as lowercase
* possible age 1 - 150

Steps:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON

{

"firstName": "Marek",

"lastName": "Benda",

"email": "[marek.benda@seznam.cz](mailto:marek.benda@seznam.cz)",

"age": "22"

}

Expected result:

1. Status code 201created
2. Response

{

"id": xxxx,

"firstName": "marek",

"lastName": "benda",

"email": "marek.benda@seznam.cz",

"age": 22

}

Actual result:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON
3. {

"firstName": "Marek",

"lastName": "Benda",

"email": "marek.benda@seznam.cz",

"age": 22

}

1. SEND

Response:

1. Status 200 OK
2. {

"id": 1278,

"firstName": "Marek",

"lastName": "BENDA",

"email": "marek.benda@seznam.cz",

"age": 22

}

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Test failed:

* The new student was added but current status 200 OK is incorrect.
* “First Name” was not changed to “LowerCase”.
* “Last Name” was changed to “Upper Case” instead of to “LowerCase”.

Expected result:

Status should be 201 Created.

{

"id": 1278,

"firstName": "marek",

"lastName": "benda",

"email": "marek.benda@seznam.cz",

"age": 22

}

## 008 Abstract: Create new student when all input data are the same as data of already existed student

Data preparation:

A screenshot of a computer

Description automatically generated

POST – create new student Marek Benda, [marek.benda@seznam.cz](mailto:marek.benda@seznam.cz), 22 let

Steps:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON

{

"firstName": "Marek",

"lastName": "Benda",

"email": "[marek.benda@seznam.cz](mailto:marek.benda@seznam.cz)",

"age": "22"

}

Expected result:

1. Status code 400 Bad request
2. Response

Message: The student already exists in the database.

Actual result:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON
3. {

"firstName": "Marek",

"lastName": "Benda",

"email": "marek.benda@seznam.cz",

"age": 22

}

1. SEND

Response:

1) Status 200 OK

2) Response

{

"id": 1287,

"firstName": "Marek",

"lastName": "BENDA",

"email": "marek.benda@seznam.cz",

"age": 22

}

A screenshot of a computer

Description automatically generated3) Check database

A screenshot of a computer

Description automatically generated

Test failed. Another student has been saved with all the same data.

It doesn’t make sense to create duplicity student.

Expected result:

* 1. Status 400 Bad Request
  2. Response

Message: Student already exists. Minimally email address must be different.

I would suggest adding a new condition: minimally an email address must be different.

## 009 Abtract: Add a new student with some missing data

Data Preparation:

POST – create new student: firstName missing, lastName Vopice, email missing, 78 let

Conditions:

* all data are manatory (firstName, lastName, email, age)
* all data are saved as lowercase
* possible age 1 – 150

Steps:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON

{

"lastName": "Vopice",

"age": "78"

}

Expected result:

Response:

1. Status 400 Bad Request

Executed steps:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON

{

"lastName": "Vopice",

"age": "78"

}

1. Press button SEND

Actual result:

1. Status 500 Internal Server Error
2. {

"timestamp": "2024-06-25T10:05:43.408+00:00",

"status": 500,

"error": "Internal Server Error",

"message": "",

"path": "/api/v1/students/"

}

A screenshot of a computer

Description automatically generated

1. Check database – student was not added to the database.

Test failed.

Expected result:

Response:

1. Status 400 Bad Request
2. Message: New student was not created due to missing mandatory data.

**B)**

Steps:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON

{

"firstName": "",

"lastName": "Vopice",

"email": "",

"age": "78"

}

Expected result:

Response:

1. Status 400 Bad Request

Executed steps:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON

{

"firstName": "",

"lastName": "Vopice",

"email": "",

"age": "78"

}

1. Press button SEND

Response:

1. Status code 200 OK
2. Response:

{

"id": 1292,

"firstName": "",

"lastName": "VOPICE",

"email": "",

"age": 78

}

**A screenshot of a computer

Description automatically generated**

1. Check database

**A screenshot of a computer

Description automatically generated**

Test failed. Incorrect status 200 OK, instead of 400 Bad Request. Student was added to the database, but it shouldn’t have been added due to some missing mandatory data (firstName, email).

Expected result:

1. Status code 400 Bad request
2. Message: New student was not created due to missing mandatory data.

## 010 Abstract: Add a new student with incorrect email address

Data Preparation:

POST – create new student Václav Mrkvička, [Vaclav.mrkvicka.seznam.cz](mailto:marek.benda@seznam.cz), 33 let

Conditions:

* all data are manatory (firstName, lastName, email, age)
* all data are saved as lowercase
* possible age 1 – 150

Steps:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON

{

"firstName": "Václav",

"lastName": "Mrkvička",

"email": "[Vaclav.mrkvicka.seznam.cz](mailto:marek.benda@seznam.cz)",

"age": "33"

}

Expected result:

1. Status code 400 Bad Request
2. Response

New student was not added to the database due to incorrect data: email address is not in correct form.

Executed Steps:

1. POST <http://108.143.193.45:8080/api/v1/students/>
2. Body + raw + JSON

{

"firstName": "Václav",

"lastName": "Mrkvička",

"email": "[Vaclav.mrkvicka.seznam.cz](mailto:marek.benda@seznam.cz)",

"age": "33"

}

1. Press button SEND

Actual result:

Response:

1. Status 200 OK

{

"id": 1293,

"firstName": "Václav",

"lastName": "MRKVIČKA",

"email": "Vaclav.mrkvicka.seznam.cz",

"age": 33

}

*A screenshot of a computer

Description automatically generated*

1. Checked database

*A screenshot of a computer

Description automatically generated*

Test failed.

New student was added to the database even though email address is not proper form.

Expected result:

1. Status code 400 Bad Request.
2. Message: Student was not added due to incorrect email address.