Exercise 1

1. GET / POST methods + JSON. (3p)

   Create the WSGI application with allstudents variable holding the data
   app – flask applikaatio
   allstudents

```python
from flask import Flask, request, jsonify, make_response

app = Flask(__name__)

allstudents = [
    {
        "student_number": 123123,
        "name": "Alice Brown",
        "credits": 130,
        "degree": "it"
    },
    {
        "student_number": 111222,
        "name": "Bob Jones",
        "credits": 157,
        "degree": "it"
    },
    {
        "student_number": 333444,
        "name": "Richard Brown",
        "credits": 57,
        "degree": "machine"
    }
]
```
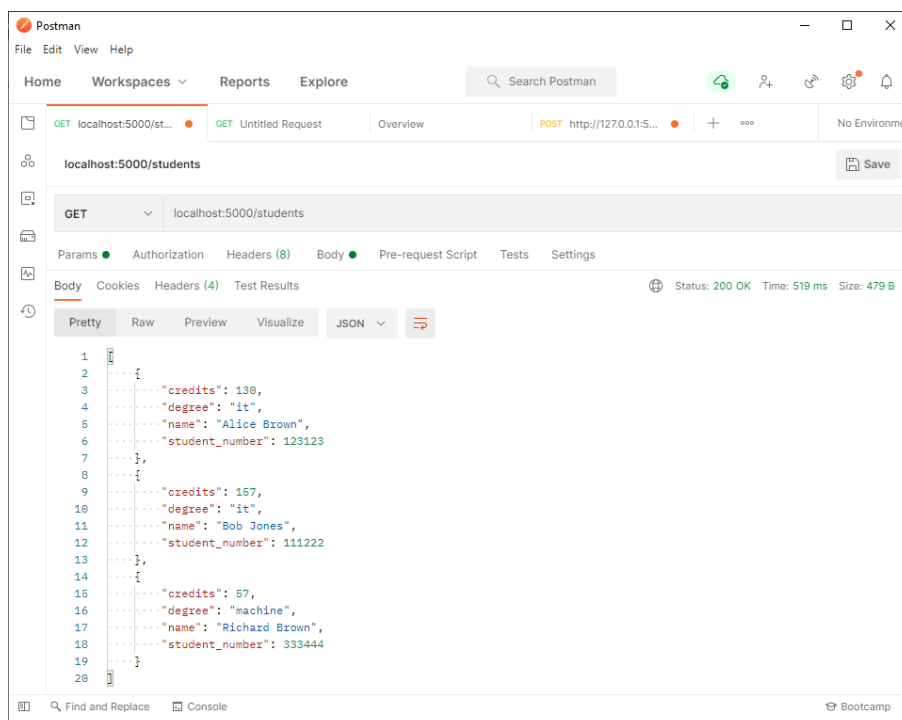
   Implement the following API and test the api routes with POSTMAN

   a. Method: GET
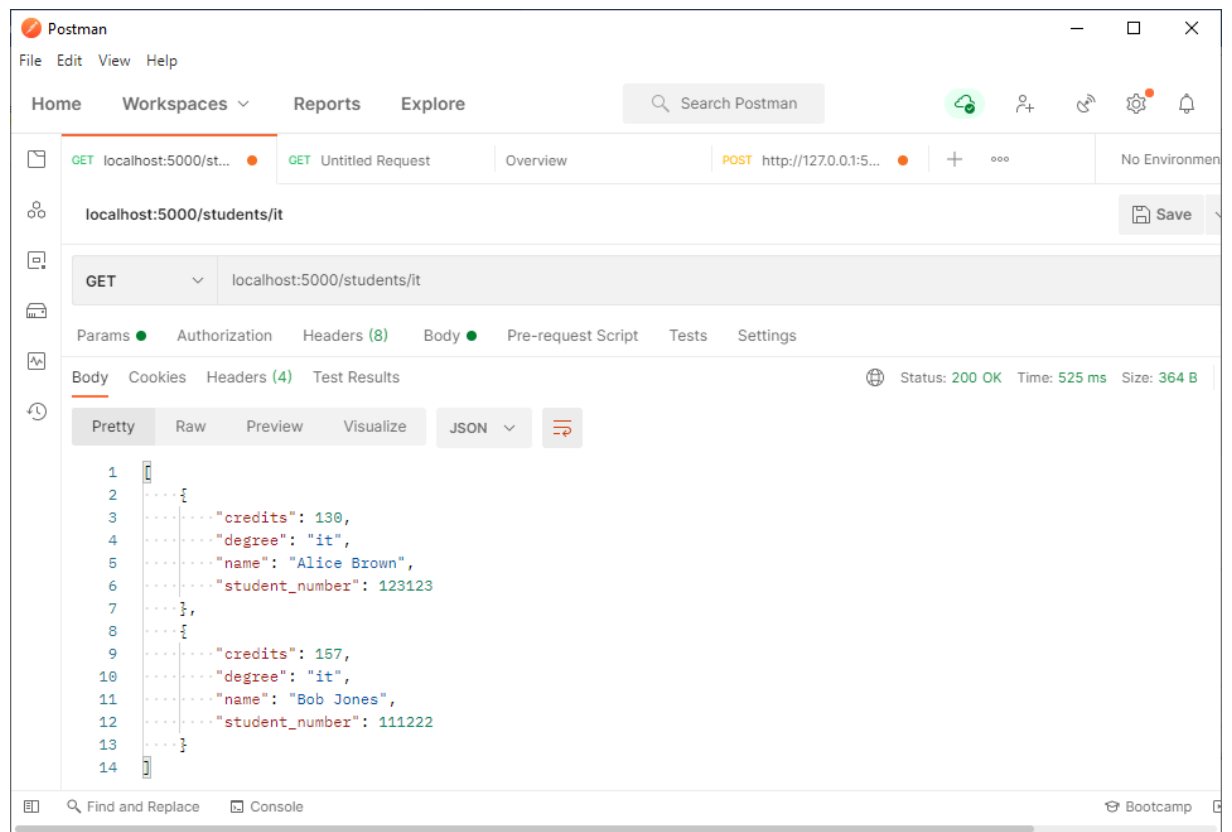   Route: /students
      Response: return all the students in json format, status code is 200.

b. Method: GET
Route: /students/<degree>
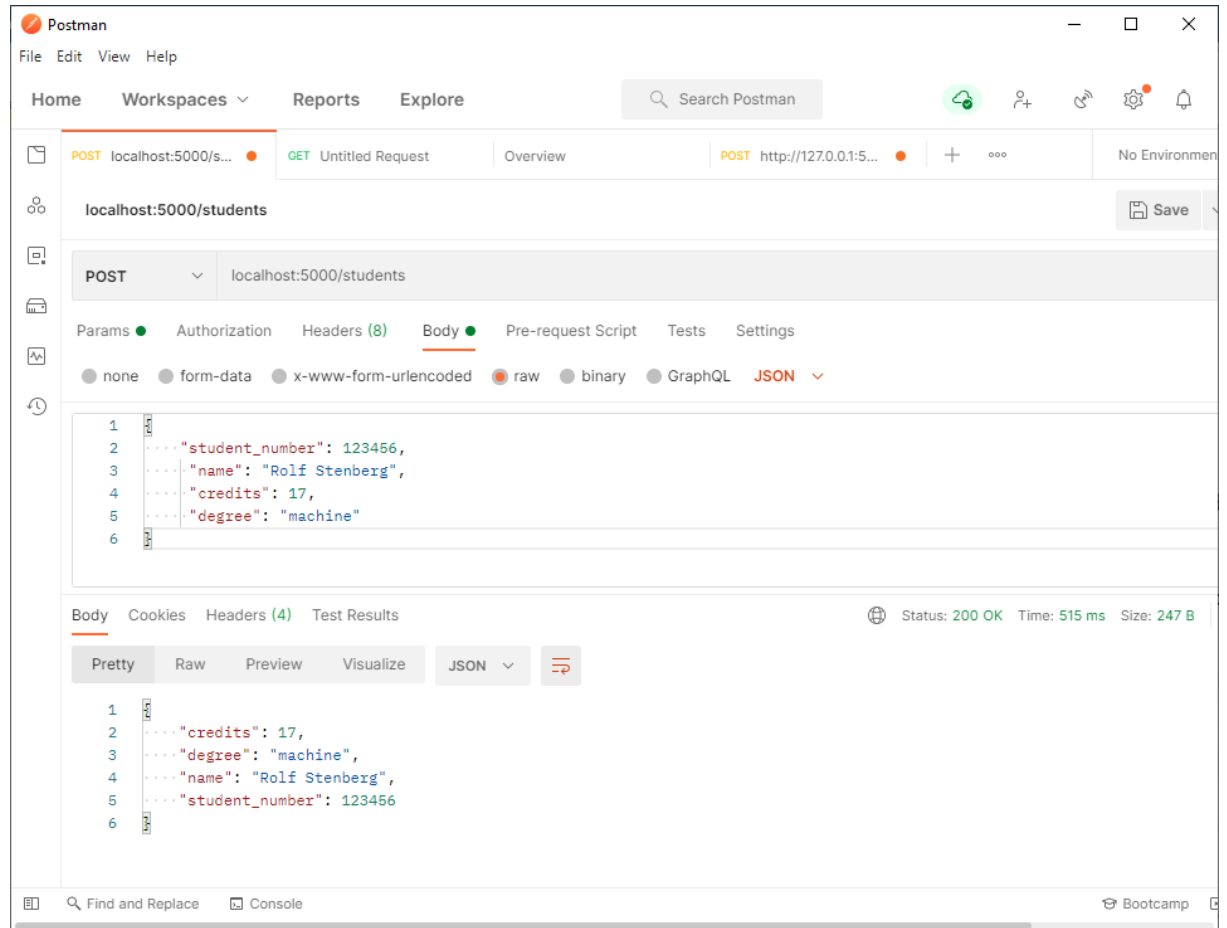Response: Returns the students whose degree mathes with the parameter

```json
[
    {
        "credits": 130,
        "degree": "it",
        "name": "Alice Brown",
        "student_number": 123123
    },
    {
        "credits": 157,
        "degree": "it",
        "name": "Bob Jones",
        "student_number": 111222
    }
]
```

c.   Method: POST
Route: /students
Data: Send the json data in request body

If the data is not according to json format return the error message with status code 400.
"Request must contain data in json format"

Otherwise the student should be added to the list and the added student is returned in the response
with status code 200

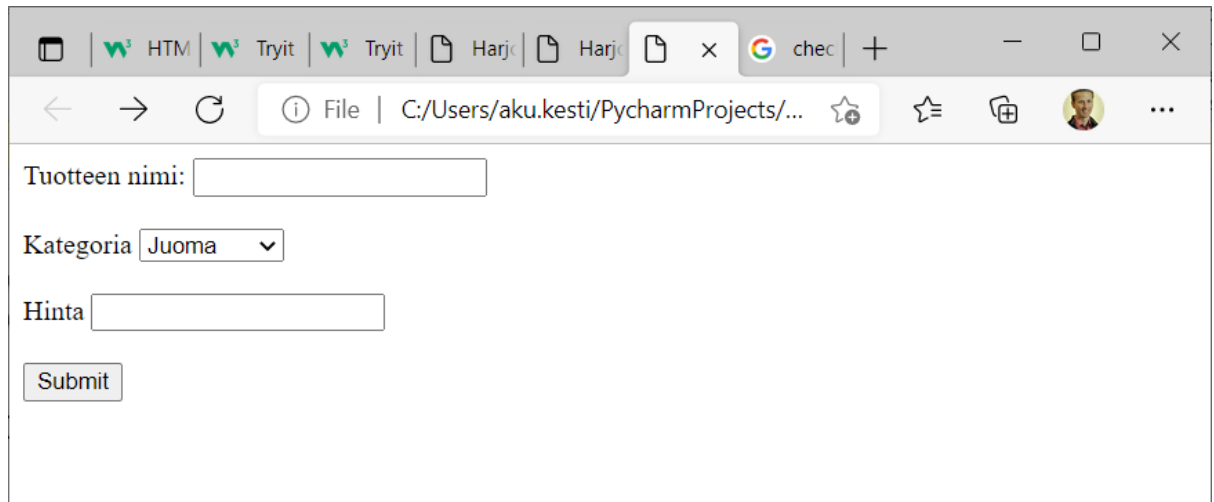2. Create the similar form as below:

   Tuotteen nimi = Product name
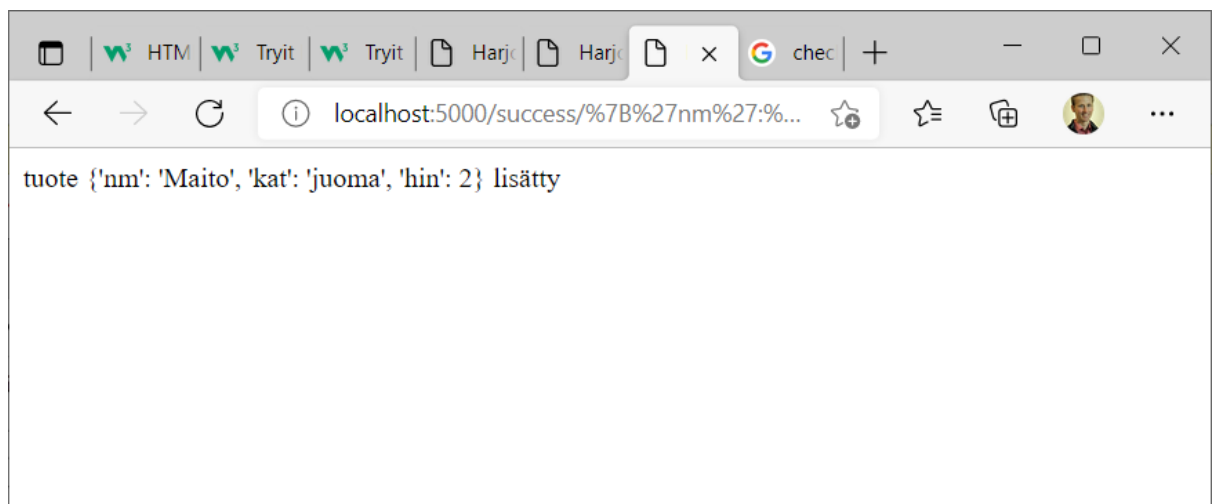   Kategoria = Category
   Hinta = Price

   If everything goes fine show the product added as in the figure 2.

   If name or price is not given then show the error message like in figure 3. (MAX 2p)
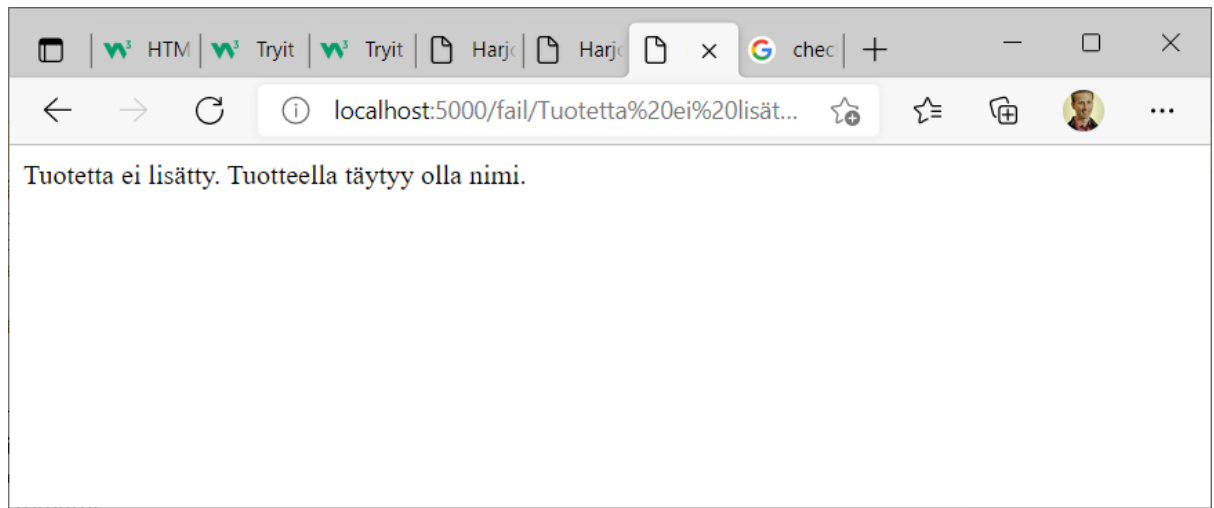

Kuva 1

Kuva 2

Tuotetta ei lisätty. Tuotteella täytyy olla nimi.

Kuva 3