

Problem: 2 Application - 'math api'

Pre-requisites: a. Python3

b. Flask

c. Curl

Considerations:

- a. Quantifier is always positive, negative quantifier doesn't return anything
- b. Operations are performed on non-empty lists. Empty list doesn't return anything.

a. What is built

A Web application to perform different math functions like min, max, avg, avg, median and percentile. Web application is created with the help of Flask framework and tested used Curl

b. How is it built

After having the programming environment setup, flask web application is created under the file name 'web.py'. All the 5 functions are executed from the same file.

Firstly, have imported the needed packages as below:

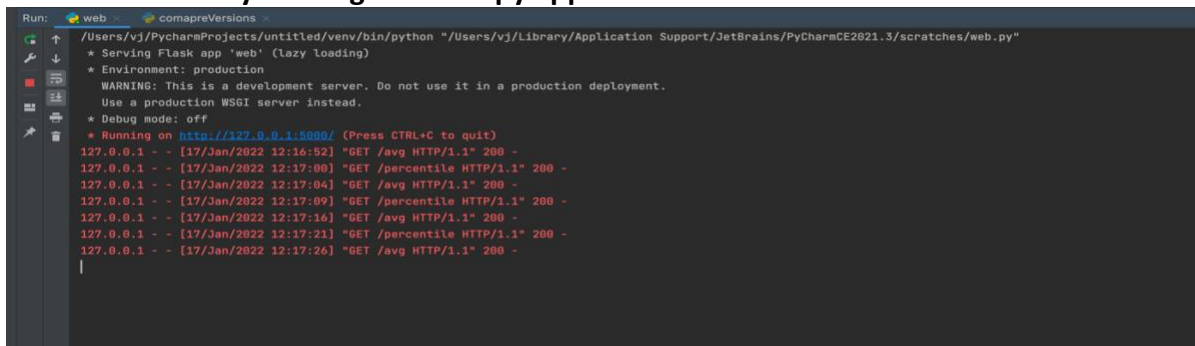
```
import statistics
import numpy as np

from flask import Flask
from flask import request
from heapq import nsmallest
from heapq import nlargest
```

1. Used heapq module to find the N minimum number of elements from the list and N maximum number of elements from the list
2. Used Numpy library to compute the qth percentile of the list elements
3. Used Statistics library to compute median of list elements

c. Testcases

a. Start by running the Web.py app which initiates the Flask as below –



```
Run: web comapreVersions
/Users/vj/PycharmProjects/untitled/venv/bin/python "/Users/vj/Library/Application Support/JetBrains/PyCharmCE2021.3/scratches/web.py"
* Serving Flask app 'web' (lazy loading)
* Environment: production
WARNING: This is a development server. Do not use it in a production deployment.
Use a production WSGI server instead.
* Debug mode: off
* Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
127.0.0.1 - - [17/Jan/2022 12:16:52] "GET /avg HTTP/1.1" 200 -
127.0.0.1 - - [17/Jan/2022 12:17:00] "GET /percentile HTTP/1.1" 200 -
127.0.0.1 - - [17/Jan/2022 12:17:04] "GET /avg HTTP/1.1" 200 -
127.0.0.1 - - [17/Jan/2022 12:17:09] "GET /percentile HTTP/1.1" 200 -
127.0.0.1 - - [17/Jan/2022 12:17:16] "GET /avg HTTP/1.1" 200 -
127.0.0.1 - - [17/Jan/2022 12:17:21] "GET /percentile HTTP/1.1" 200 -
127.0.0.1 - - [17/Jan/2022 12:17:26] "GET /avg HTTP/1.1" 200 -
```

i. Testing using Curl

After Web.py file starts running – run the CURL commands to test and see the results of the Math Functions

Min Function

```
terminal Local + v
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [3,2,4,1], "quantifier": 3}' -H "Content-Type: application/json" -X GET http://localhost:5000/max
[4, 3, 2]
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [3,3,3,3], "quantifier": 3}' -H "Content-Type: application/json" -X GET http://localhost:5000/max
[3, 3, 3]
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [3,2,4,1], "quantifier": -3}' -H "Content-Type: application/json" -X GET http://localhost:5000/min
[]
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [3,2,4,1], "quantifier": 0}' -H "Content-Type: application/json" -X GET http://localhost:5000/min
[]
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [3,2,4,1], "quantifier": 2}' -H "Content-Type: application/json" -X GET http://localhost:5000/min
[1, 2]
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [], "quantifier": 2}' -H "Content-Type: application/json" -X GET http://localhost:5000/min
[]
(env) VJs-MacBook-Pro:untitled vj$
```

Max Function

```
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [3,2,4,1], "quantifier": 3}' -H "Content-Type: application/json" -X GET http://localhost:5000/max
[4, 3, 2]
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [3,3,3,3], "quantifier": 3}' -H "Content-Type: application/json" -X GET http://localhost:5000/max
[3, 3, 3]
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [3,3,3,3], "quantifier": -1}' -H "Content-Type: application/json" -X GET http://localhost:5000/max
[]
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [3,5,8,10], "quantifier": 2}' -H "Content-Type: application/json" -X GET http://localhost:5000/max
[10, 8]
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [], "quantifier": 3}' -H "Content-Type: application/json" -X GET http://localhost:5000/max
[]
(env) VJs-MacBook-Pro:untitled vj$
```

Avg Function

```
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": []}' -H "Content-Type: application/json" -X GET http://localhost:5000/avg
0
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [1,2,3,4]}' -H "Content-Type: application/json" -X GET http://localhost:5000/avg
2.5
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [-1,-3,-5,-10]}' -H "Content-Type: application/json" -X GET http://localhost:5000/avg
-4.75
(env) VJs-MacBook-Pro:untitled vj$
```

Median Function

```
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [1,1,2,4]}' -H "Content-Type: application/json" -X GET http://localhost:5000/median
1.5
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": []}' -H "Content-Type: application/json" -X GET http://localhost:5000/median
Empty List
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [-5,-9,2,4]}' -H "Content-Type: application/json" -X GET http://localhost:5000/median
-1.5
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [-10,-8,-5,-6]}' -H "Content-Type: application/json" -X GET http://localhost:5000/median
-7.0
(env) VJs-MacBook-Pro:untitled vj$
```

Percentile function

```
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [20,2,7,1,34], "quantifier": 50}' -H "Content-Type: application/json" -X GET http://localhost:5000/percentile
7.0
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [20,2,7,1,34], "quantifier": 2}' -H "Content-Type: application/json" -X GET http://localhost:5000/percentile
1.08
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [20,2,7,1,34], "quantifier": 100}' -H "Content-Type: application/json" -X GET http://localhost:5000/percentile
34.0
(env) VJs-MacBook-Pro:untitled vj$ curl -w "%n" -d '{"list": [20,2,7,1,34], "quantifier": 0}' -H "Content-Type: application/json" -X GET http://localhost:5000/percentile
1.0
(env) VJs-MacBook-Pro:untitled vj$
```

ii. Testing using Postman (three screenshots are attached)

Percentile Function with 0 quantifier

The screenshot shows the Postman interface for a GET request to `http://localhost:5000/percentile`. The request body is a JSON object: `{ "list": [20, 2, 7, 1, 34], "quantifier": 0 }`. The response is `1.0`.

```
1 { "list": [20, 2, 7, 1, 34], "quantifier": 0 }
```

```
1 1.0
```

Avg function

The screenshot shows the Postman interface for a GET request to `http://localhost:5000/avg`. The request body is a JSON object: `{ "list": [-1, -3, -5, -10] }`. The response is `-4.75`.

```
1 { "list": [-1, -3, -5, -10] }
```

```
1 -4.75
```

Min function

The screenshot displays the Postman application interface. At the top, there's a search bar and navigation links like 'Sign In' and 'Create Account'. Below this, a yellow banner indicates 'Working locally in Scratch Pad. Switch to a Workspace'. The main area shows a list of requests, with the selected one being a GET request to 'http://localhost:5000/min'. The request details are visible, including the method 'GET', the URL, and the body set to 'raw' JSON format. The body content is:

```
1 { "list": [3,2,4,1], "quantifier": 3 }
```

. The response section at the bottom shows a status of '200 OK', a time of '7 ms', and a size of '161 B'. The response body is displayed in 'Pretty' HTML format as:

```
1 [1, 2, 3]
```

.