CS 251 InLab5 - Python Basics

Total score - 30
Resources -

1. Basic file reading -
    a. [Python File I/O: Read and Write Files in Python](#)
    b. [JSON encoder and decoder — Python 3.10.6 documentation](#)
    c. [Python eval() built-in-function](#)
2. Classes and objects (OOPS) in Python -
    a. [Python Classes and Objects - GeeksforGeeks](#)
    b. [Python Classes and Objects [With Examples]](#)
    c. [9. Classes — Python 3.10.6 documentation](#)
    d. [Python Classes/Objects](#)
    e. [Python Static Methods Explained Clearly By Practical Examples](#)
3. https://en.wikipedia.org/wiki/Sieve_of_Eratosthenes
4. https://www.geeksforgeeks.org/python-lambda-anonymous-functions-filter-map-reduce/
5. https://www.geeksforgeeks.org/python-map-function/

Instructions -

Python Basics (Inlab)   Updated automatically every 5 minutes

raising the given Exception class will result in loss of marks from the autograder.

2. Resources for this lab are in inlab5-resources.tar.gz. Untar it using

   tar -xvzf inlab5-resources.tar.gz

3. The .tar file contains the following files - exception.py, q1.py, q1_runner.py, student.py, test_case_q1.json, q2.py

Questions -

Q.1 Over your Software Engineering career you would face situations that force you to follow the best coding practices. In this problem, you are to implement two such coding practices - Factory Pattern (with single inheritance, you can safely ignore inheritance, it is not needed to solve this) and Singleton Pattern.

(a) Factory Pattern **encapsulates** the mechanism of instantiating a class and abstracts out a simple callable function to get an instance of the class.

Python Basics (Inlab)

Updated automatically every 5 minutes

provide an implementation just for the constructor. Additionally, you are to implement a static method in q1.py. It takes in a path to a JSON file. It should read the file, extract relevant information and instantiate the class from student.py and return the object. Your code would be checked as-

    student_obj = StudentBuilder.build_student_object('path_to_file.json')

The JSON file will contain the following fields-
- name
- age (assume highest age of student is 35)
- cgpa (on a 0-10 scale, both inclusive)
- gender (only 'Male', 'Female', 'Non-Binary', 'Prefer Not To Say' are valid)
- home_town

A sample file will be provided to test your code in the lab. Additionally, a helper file is provided q1_runner.py, which you can use to test your code. You can run it as -

python q1_runner.py

Please note that there will be no invalid files - all input test cases **will** be correctly formatted. However, there might be cases where the file can have missing values, or it can have invalid values (age for name field, gender for age field). Brainstorm on what else can be invalid (Can age be negative? ;)). **[10 Marks]**

Python Basics (Inlab)

Updated automatically every 5 minutes

implement the Singleton Pattern within the builder method.   **[5 Marks]**

Q.2   Write a Python function which implements the [Sieve of Eratosthenes](#) method for computing primes up to a specified number (Less than or equal to the given number)   **[10 Marks]**

You should collect all the prime numbers in a list and plot the numbers on Y=X line i.e, for plotting the number the x-coordinates and y-coordinates should be the same and each coordinate corresponds to one of the prime numbers from the list you got after finding all the prime numbers till given integer N.   **[5 Marks]**

Important: Your code must accomplish this using a combination of Lambda functions, map,reduce and filter.
You can use at most one loop for 90% marks on this problem. Full marks only if you do it without any loop. For any more loops than one, no marks.

For marking, your method for prime numbers will be checked which should be the **Sieve of Eratosthenes** and the graph for the primes numbers.
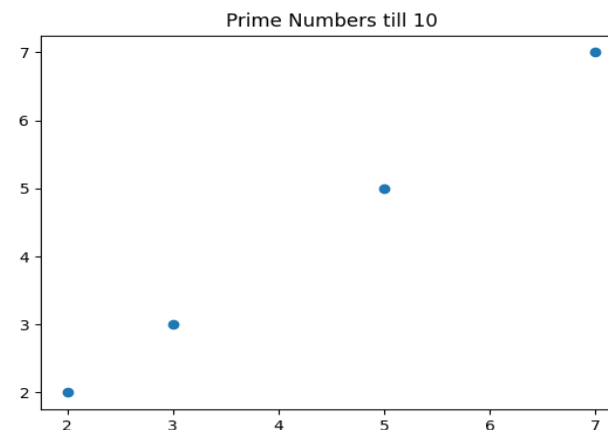
- Running your q2.py script:

Python Basics (Inlab)

console output should look like this.

List of primes =
[2,3,5,7]

And the corresponding graph should look like the following :

Submission Instructions

The following is what your submission directory must look like. Please follow the submission format strictly.

The submission directory must be named **<roll_no>_inlab5**

Compress it into a tarball using the following command

Python Basics (Inlab)

```
├── exception.py
├── q2.py
```

Note - You **do not** need to include q1_runner.py and the json
file with your submission!!