Week 4: Assignment 2
Guess the Topper of a Class

[*Last Updated on:* ***26th April, 2021, 14:00 Hrs***]

Aim

Your big brother is a tutor in an academy teaching Engineering subjects. Last week he conducted an online test in class and now has a list of student names and their scores.

He wants to find the name of student with maximum score and give a shout-out to him/her on social media platforms of the academy to appreciate his/her effort and motivate others as well.

But he doesn't want to find the name manually and since you are well versed with Python, he asked you to perform this task on his behalf.

As per the knowledge of your brother in Python, he provided a starting point by giving you a skeleton code stub.

Given

Two files are provided to solve this assigment.

- Skeleton program file: `assignment2.py`
  - The skeleton consists of two functions which you have to modify:
    - `readScoreSheet()`
    - `getTheTopper()`
- Sample CSV file: `week4_assignment2_sample.csv`
  - The CSV file has 2 columns: `name` and `marks` .

Procedure

- Open the skeleton program file, `assignment2.py` .

- You will notice pre-written comments included in skeleton program for your assistance to solve the assignment.

- Two functions to modify are:

  - `readScoreSheet()`

Function Name	`readScoreSheet()`
Purpose	Reads the input CSV file of Score Sheet and creates a mapping of student name with his/her marks.
Input Arguments	`file_name` : [***str***] CSV file name of Score Sheet
Output Arguments	`name_score_mapping` : [***dict***] Mapping of each student's name and his/her marks as { Key : Value } pair
Example Call	***name_score_mapping = readScoreSheet(csv_file_name)***

  - `getTheTopper()`

Function Name	`getTheTopper()`
Purpose	Get the name of student with maximum score in the given mapping of student and their scores.
Input Arguments	`mapping_dict` : [***dict***] Mapping of each student's name and his/her marks as { Key : Value } pair
Output Arguments	`name_with_max_score` : [***str***] Name of student with maximum score
Example Call	***topper_name = getTheTopper(name_score_mapping)***

- To run and debug your solution, type the below command in Terminal:

```
$ python3 assignment2.py
```

  This command will run the Python script `assignment2.py` .

- Refer the **Expected Output** section below and debug your code to get the correct output.

> **Hint:** *Make use of* ***lambda*** *function to find the student name of topper for iterating over dictionary quickly.*

Expected Output

- The provided sample CSV file, `week4_assignment2_sample.csv` consists data with fields ***name***, ***marks***.

- The content of this CSV file is shown below:

```
name,marks
Harv,20
Virge,52
Teresita,93
Maire,9
```

```
Jenica,57
Kermie,32
Chelsie,11
Emilio,95
Wynne,14
Kelley,66
```

- The expected output of program `assignment2.py` i.e., to print the name of student with maximum score is shown below:

```
{
    'Harv': 20.0, 'Virge': 52.0, 'Teresita': 93.0, 'Maire': 9.0, 'Jenica': 57.0,
    'Kermie': 32.0, 'Chelsie': 11.0, 'Emilio': 95.0, 'Wynne': 14.0, 'Kelley': 66.0
}
Emilio
```

- The first four lines above are the output of `readScoreSheet` function i.e. variable `name_score_mapping` and,

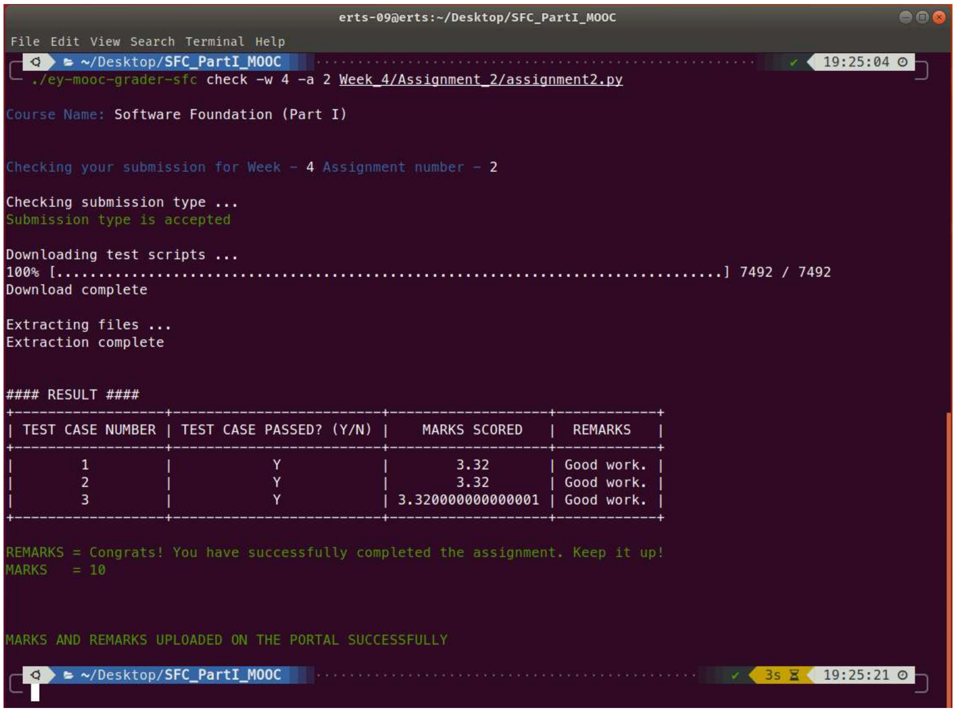
  the last line is the output of `getTheTopper` function i.e. variable `topper_name` .

Grading and Submission Instructions

- Navigate to the folder where the ***ey-mooc-grader-sfc*** application resides.
- To grade your solution, run the `check` command of the application as follows:

```
$ ./ey-mooc-grader-sfc check -w 4 -a 2 Week_4/Assignment_2/assignment2.py
```

- This will run your program `assignment2.py` against random test cases and grade it. Marks and appropriate remarks will be provided as shown in Figure 1.
- Your program file `assignment2.py` , marks scored and remarks will get uploaded to the MOOC portal.

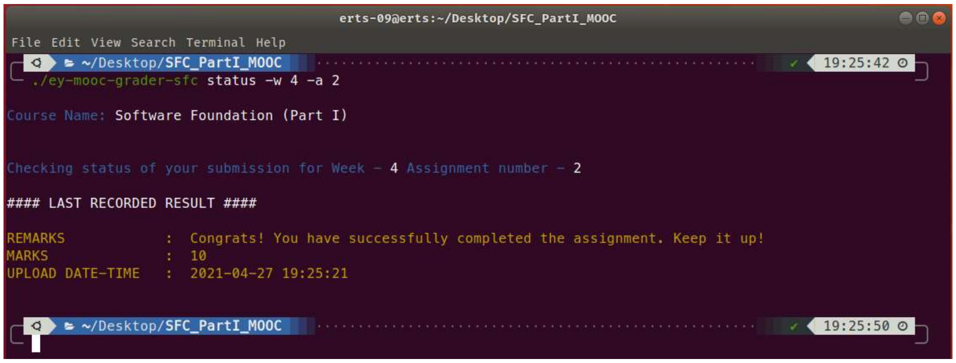

erts-09@erts:~/Desktop/SFC_PartI_MOOC
File Edit View Search Terminal Help

```
 ~/Desktop/SFC_PartI_MOOC ................................................ ✓ 19:25:04
 ./ey-mooc-grader-sfc check -w 4 -a 2 Week_4/Assignment_2/assignment2.py

Course Name: Software Foundation (Part I)


Checking your submission for Week - 4 Assignment number - 2

Checking submission type ...
Submission type is accepted

Downloading test scripts ...
100% [................................................................................] 7492 / 7492
Download complete

Extracting files ...
Extraction complete


#### RESULT ####
+------------------+--------------------------+-------------------+------------+
| TEST CASE NUMBER | TEST CASE PASSED? (Y/N) |    MARKS SCORED    |  REMARKS   |
+------------------+--------------------------+-------------------+------------+
|        1         |            Y             |        3.32       | Good work. |
|        2         |            Y             |        3.32       | Good work. |
|        3         |            Y             | 3.320000000000001 | Good work. |
+------------------+--------------------------+-------------------+------------+

REMARKS = Congrats! You have successfully completed the assignment. Keep it up!
MARKS   = 10



MARKS AND REMARKS UPLOADED ON THE PORTAL SUCCESSFULLY

 ~/Desktop/SFC_PartI_MOOC ........................................ ✓ 3s ⧗ 19:25:21
```

Figure 1: Output of running check command for Week 4 Assignment 2

- You can verify this by running the **`status`** command of the application as given below, refer Figure 2.

```
$ ./ey-mooc-grader-sfc status -w 4 -a 2
```


erts-09@erts:~/Desktop/SFC_PartI_MOOC
File Edit View Search Terminal Help

```
 ~/Desktop/SFC_PartI_MOOC ........................................... ✓ 19:25:42
 ./ey-mooc-grader-sfc status -w 4 -a 2

Course Name: Software Foundation (Part I)

Checking status of your submission for Week - 4 Assignment number - 2

#### LAST RECORDED RESULT ####

REMARKS           :  Congrats! You have successfully completed the assignment. Keep it up!
MARKS             :  10
UPLOAD DATE-TIME  :  2021-04-27 19:25:21

 ~/Desktop/SFC_PartI_MOOC ........................................... ✓ 19:25:50
```

Figure 2: Output of running status command for Week 4 Assignment 2

References

- Official Python documentation of CSV module
- Blog on Python CSV Module by RealPython
- Blog on For Loops by RealPython
- Blog on Python Dictionary by Programiz
- Blog on Python Dictionary by RealPython
- Blog on Lambda Functions by RealPython

All The Best!