X

 (https://swayam.gov.in)  (https://swayam.gov.in/nc_details/NPTEL)

gauravsharma727545@gmail.com

NPTEL (https://swayam.gov.in/explorer?ncCode=NPTEL) » Programming, Data Structures And Algorithms Using Python (course)

Announcements (announcements)

About the Course (https://swayam.gov.in/nd1_noc19_cs40/preview) Ask a Question (forum)

Progress (student/home) Mentor (student/mentor)

Course outline

- **How to access the portal**
- **Week 1: Introduction**
- **Week 1 Quiz**
- **Week 2: Basics of Python**
- **Week 2 Quiz**
- **Week 2 Programming Assignment**
- **Week 3: Lists, inductive function**

Online Test 2, Question 8

Due on 2019-09-26, 22:00 IST

Instructions

This is the second of two online programming tests.

- These tests account for 25% of the total evaluation for the course.
- The duration of the test is 2 hours.
- The first test was from 9:30-11:30 am and the second is from 8:00-10:00 pm, on Thursday, 26 September 2019.
- You can attempt either of the tests. The best score will be counted..

Note: In this question, you have to write a Python function. Your function should return the value specified in the problem description. Do not print any messages or diagnostic information. Your code wll be evaluated automatically by comparing your program's output with the expected output, so any spurious output from your program will cause your answer to be reported as wrong. You can assume that inputs to your functions will be of the correct type, as specified in the question.

There are some "public" test cases where you can see how your program does when you use "Compile and Run". Finally, you should "Submit" your code for evaluation. Your solution will be checked against "private" test cases, which you cannot see. You will get a score on 100 based on how many private test cases you solve correctly.

Question 8

Write a Python function `maxaverage(l)` that takes a list of pairs of the form (name,score) as argument, where name is a string and score is an integer. Each pair is to be interpreted as the score of the named player. For instance, an input of the form [('Kohli',73),('Ashwin',33),('Kohli',7), ('Pujara',122),('Ashwin',90)] represents two scores of 73 and 7 for Kohli, two scores of 33 and 90 for Ashwin and one score of 122 for Pujara. Your function should compute the players who have the highest average score (average = total across all scores for that player divided by number of entries) and return the list of names of these players as a list, sorted in alphabetical order. If there is a single player, the list will contain a single name.

For instance, `maxaverage([('Kohli',73),('Ashwin',33), ('Kohli',7),('Pujara',122),('Ashwin',90)])` should return `['Pujara']` because the average score of Kolhi is 40 (80 divided by 2), of Ashwin is 61.5 (123 divided by 2) and of Pujara is 122 (122 divided by 1), of which 122 is the highest.

Private Test cases used for evaluation	Input	Expected Output	Actual Output	Status
Test Case 1	maxaverage([('Kohli',73),('Ashwin',33),('Kohli',7),('Pujara',142),('Ashwin',90)])	['Pujara']\n	['Pujara']\n	Passed
Test Case 2	maxaverage([('Kohli',73),('Ashwin',33),('Kohli',7),('Pujara',100),('Pujara',25),('Pujara',35),('Ashwin',109)])	['Ashwin']\n	['Ashwin']\n	Passed
Test Case 3	maxaverage([('Kohli',73)])	['Kohli']\n	['Kohli']\n	Passed
Test Case 4	maxaverage([('Kohli',73),('Ashwin',33),('Kohli',69),('Pujara',102),('Pujara',40),('Ashwin',109)])	['Ashwin', 'Kohli', 'Pujara']\n	['Ashwin', 'Kohli', 'Pujara']\n	Passed

Due Date Exceeded.
4 out of 4 tests passed.
You scored 100.0/100.

Your last recorded submission was :

```
from collections import defaultdict
import statistics

def maxaverage(l):
    result = defaultdict(list)
    for i, j in l:
        result[i].append(j)
    result = dict(result)
    aggregate_score = {}
    for k, v in result.items():
            aggregate_score.update({k:statistics.mean(v)})
    itemMaxValue = max(aggregate_score.items(), key=lambda x: x[1])
    listOfKeys = list()
    for key, value in aggregate_score.items():
        if value == itemMaxValue[1]:
            listOfKeys.append(key)
    listOfKeys.sort()
    return listOfKeys
import ast

def tolist(inp):
  inp = ast.literal_eval(inp)
  return (inp)

fncall = input()
lparen = fncall.find("(")
rparen = fncall.rfind(")")
```

(/noc19_cs40/progassig
name=125)

- Online Test 2, Question 6 (/noc19_cs40/progassig
name=126)
- Online Test 2, Question 7 (/noc19_cs40/progassignment?
name=127)
- **Online Test 2, Question 8 (/noc19_cs40/progassignment?
name=128)**
- Online Programming Test 2, 26 Sep 2019, 20:00-22:00 (unit?
unit=111&lesson=129)

```
fname = fncall[:lparen]
farg = fncall[lparen+1:rparen]

if fname == "maxaverage":
  arg = tolist(farg)
  print(maxaverage(arg))

```