

CSCA20 - Final Exam

Unhappy Tapirs Scrapped for Convenience

Academic Integrity

While this is a take-home assignment, it is being treated as an exam. That means that everything you submit must be entirely your own work. In particular, please be aware of the following points:

- You MAY NOT discuss or share your work with anyone
- You MAY NOT look at anyone else's solutions
- You MAY NOT use any external resources (websites, books, tutoring services, etc)
- You MAY ask clarification questions on Piazza
- You MAY NOT post any of your own code on Piazza
- You MAY use any code posted to Quercus or Piazza by the course instructors or TAs
- You MAY add the comment 'I read the handout' as the first line of your code to receive a bonus mark
- You MAY use any code you already wrote for previous labs

Submission

You must submit your ipynb file to quercus by 7:00PM Toronto time on December 17.

UTSC

You've done a really good job with your first two iterations of the UTSC app. But now, to make things simpler, we won't be starting from scratch (pun 100% intended), instead you will only have to complete some pre-defined functions that the app needs in order to complete its tasks.

There are 3 main pieces of functionality that we need to be able to complete: Finding all values between two cutoffs, calculating the top three entries (for gold, silver and bronze respectively), and then finally deciding on the champion(s) given all the data we've collected. But we will need to perform the same functionality on different types of inputs.

Starter code with function definitions and some simple testing code have been provided for you. Your job is to complete the functions in the code (you will also likely want to add more testing code, but we won't be marking that) to the following specifications.

All code must be properly tested and documented (internal and external comments) in order to receive full marks. You may make any reasonable requirements of the input (e.g., number of elements in the list or relative values of cutoffs), but those requirements must be clearly stated in the external comments.

Lists

`list_between`

Given an input list, a lower cutoff value and an upper cutoff value, return a copy of the list only containing values between the two cutoffs.

`list_top_three`

Return a list containing the top 3 items in the input list, in order [largest, 2nd largest, 3rd largest].

Dictionaries

`dict_between`

Given a dictionary mapping names to scores, a lower cutoff value and an upper cutoff value, return a copy of the dictionary only containing entries with scores that fall between the two cutoffs.

`dict_top_three`

Given a dictionary with names mapping to scores, return a new dictionary with 'Gold' mapping to the name with the highest score, 'Silver' mapping to the name with the 2nd highest score, and 'Bronze' mapping to the name with the 3rd highest score.

CSV Files

csv_between

Given the handles of input and output csv files, a lower cutoff value and an upper cutoff value, write a properly formatted csv file to the output handle with the data from the input handle but only containing scores between the two cutoffs. The input csv file will be properly formatted with two columns, the first being a name and the second being a score.

csv_top_three

Given the handles of an input csv file and an output text file, print the winners of gold, silver, and bronze to the text files, based on the data from the input csv. The input csv file will be properly formatted with two columns, the first being a name and the second being a score.

The Championship

Given a list of input handles (the formatting and order of the input handles will be the same as shown in the starter code, file names and contents will change, but format, headers, and order of files in the input list will be consistent), your job is to print the following items about the data contained in the files:

- Top three teams by wins
- Top three teams by total points scored
- Top three players by points scored
- Top scoring player on each team

As a hint: You are not required to use SQL for this, but your life will be much easier if you do.

Marking

The marking scheme for the exam will be as follows:

- Lists 20%
- Dictionaries 30%
- CSV Files 30%
- Championship 20%
 - 5% for each element correctly reported