Python for physicists - exercise 7

Submission instructions - please read carefully:
• To be submitted by *** in the moodle (Lemida) system.
• *** files with py suffixes must be submitted - named exactly as detailed below for each exercise. That is to say that:
• Do not submit complete projects, libraries, zip files, etc., and do not submit all exercises in one file, but in separate files with the names listed below.
• Make sure that the files run and do what is needed (on a recent version of Python, 3.5 or higher).
• Use only the commands we learned in the practice.

Exercise 1. Submit it as file name: ex07vanilla.py

In this question, please do not use *numpy, pandas*, or other libraries, use only the commands of *Python*.

Write a function named ***calc*** that takes in a list of dictionaries called ***data***, along with two string parameters: ***key1*** and ***key2***. The function should return the maximum value found in the dictionaries under ***key2***, among all data in the dictionaries where the value under ***key1*** is greater than the average value of all ***key1*** values.

For example, if the dictionary list is

data = [{ 'A': 20, 'B': 30.5}, { 'A': 15, 'B ': 140}, { 'A ': 24, 'B ': 26.25}]

and you call the function ***calc***:

calc(data, 'A ', 'B ')

So, the function returns 30.5, because the average of the values in A:(20+24+15)/3 = 19.66

Therefore, the data that have a value in key1 *'**A** '* bigger than the average of all the values in *'**A** '* are the first and the third. Between these two the first has the highest B value: 30.5

Exercise 2. Submit it as file name: ex07pandas.py

Write a function named ***calc*** that takes a pandas DataFrame named ***df***, along with two string parameters: ***key1*** and ***key2***.

The function returns the maximal value in ***df*** in column ***key2***, among all the rows where the value in column ***key1*** is greater than the average of the values in column ***key1***.

For example: for ***df = pd.DataFrame(data)***, and if data is the same as in ***Exercise 1***, the function will return 30.5. Because you are asked to write a function that does the same thing as the function in ***Exercise 1*** did above, but this time, use *pandas* commands.

(The solution should be very short, unlike the solution of ***Exercise 1***.)

Exercise 3. Submit it as file name: ex07comparison.py

Make a comparison between the speeds of the functions you wrote in ***Exercise 1*** and ***Exercise 2***. To do that ***import*** the two functions you wrote from the files above, and compare the speed with ***time.time()***

(it makes sense to check a few thousand runs in a loop to get a meaningful comparison).

Choose the data you're examining, and the number of tests, but reach the state where you feel that you have made a proper comparison between the two functions.

The summary of the comparison should be recorded in the comments at the end of the file.