212 Scientific Programming with Python Tue Thurs 06/09 - 07/14 From 10:30 AM to 12:00 PM

Online Courses

Our courses are going online! As we are faced with the COVID-19 challenge, we are moving all our training courses to live streaming using Microsoft Teams/Zoom platforms until further notice.

Final Project

What to submit

Send your source code properly documented, with Comments PeopleSoft ID, in a .ipynb file

named: FirstName_LastName_assignmentNumber.ipynb to the course web site

Your source code must run without syntax errors in order to receive a passing grade for the project

Make sure you add docstrings comments in your source code that describe what the program is supposed to do. Before submitting your code, make sure you test it with different input values to ensure it works correctly.

1)

Implement function easyCrypto() that takes as input a string and prints its encryption defined as follows: Every character at an odd position i in the alphabet will be encrypted with the character at position i + 1, and every character at an even position i will be encrypted with the character at position i - 1. In other words, ‘a’ is encrypted with ‘b’, ‘b’ with ‘a’, ‘c’ with ‘d’, ‘d’ with ‘c’, and so on. Lowercase characters should remain lowercase, and uppercase characters should remain uppercase.

easyCrypto('abc')

bad

easyCrypto('Z00')

YPP

2) Kids Game

Develop a simple game that teaches kindergartners how to add single-digit numbers.

Your function game() will take an integer n as input and then ask n single-digit addition questions. The numbers to be added should be chosen randomly from the range [0, 9] (i.e.,0 to 9 inclusive). The user will enter the answer when prompted. Your function should print 'Correct' for correct answers and 'Incorrect' for incorrect answers. After n questions, your function should print the number of correct answers.

game(3)

8 + 2 =

Enter answer: 10

Correct.

6 + 7 =

Enter answer: 12

Enter answer: 12

Incorrect.

7 + 7 =

Enter answer: 14

Correct.

You got 2 correct answers out of 3

3)

In your class, many students are friends. Let's assume that two students sharing a friend must be friends themselves; in other words, if students 0 and 1 are friends and students 1 and 2 are friends, then students 0 and 2 must be friends. Using this rule, we can partition the students into circles of friends.

To do this, implement a function networks() that takes two input arguments. The first is the number n of students in the class. We assume students are identified using integers 0 through n-1. The second input argument is a list of tuple objects that define friends. For example, tuple (0, 2) defines students 0 and 2 as friends. Function networks() should print the partition of students into circles of friends as illustrated:

networks(5, [(0, 1), (1, 2), (3, 4)])

Social network 0 is {0, 1, 2}

Social network 1 is {3, 4}

4)Integrate a function by one trapezoid

An approximation to the integral of a function f(x) over an interval [a, b] can found by first approximating f(x) by the straight line that goes through the end points (a, f(a)) and (b, f(b)), and then finding the area under the straight line (which is the area of a trapezoid). The resulting formula becomes https://wikimedia.org/api/rest_v1/media/math/render/svg/ff1e6d438c8e663ed8ab18d19b011371c24b3ac7 https://en.wikipedia.org/wiki/Trapezoidal_rule

(4a) Write a function trapezint1(f, a, b) that returns this approximation to the integral. The argument f is a Python implementation f(x) of the mathematical function f(x).

(4b) Write python code showing how you would do a line plot b vs trapezint1(f, a, b), assume a=0, and function f is the math.sin

5) Detecting Trends and Foul Language in Twitter

OBJECTIVE

This sub-project will allow you to practice manipulating dictionaries and files in your python scripts.The goal is to detect the presence of foul language and keep track of trendy topics in a sample of Twitter data. You will need two files posted on the course website: twitter_data.txt and swear_words.txt.

WARNING: Some of the tweets in the sample file actually do contain swear words.

5a). Detecting Foul Language in Twitter Microblogging sites such as Twitter and Ask.fm are sometimes misused to abuse people. In this part of the assignment your task is to screen each tweet for the presence of swear words. We provide an initial list of bad words in the file named swear_words.txt. The file twitter_data.txt contains real tweets collected to study cyberbullying. Each line is a different tweet. Write a function that will read each tweet in the file, will look for swear words, and will write to a new file named potentially_offensive_tweets.txt all tweets containing foul language. Note that the sample may have repeated tweets as well as tweets in a foreign language. You may find the need to update your swear_words.txt file. That's expected, as the list is not comprehensive.

5b) Detecting Topic Trends in Twitter

One of the services Twitter provides its users is the ability to track the most popular topics. For this part of the assignment you will do something similar. Your task is to keep track of the topics identified by users with the hashtag symbol '#'. You will also need to count the

something similar. Your task is to keep track of the topics identified by users with the hashtag symbol "#". You will also need to count the frequency of the hashtags you found and provide a ranking of hashtags based on their frequency. The output of your script should be

one file, named top_hashtags.txt, with the N most popular hashtags, where N is a parameter to your function. For example, assume this is the content of your twitter_data.txt file:

#lebron best athlete of our generation

ML 5 Demos! Lots of great stuff to come! Yes, I'm excited. :) http://htmlfive.appspot.com #io2009 #googleio

At GWT fireside chat #googleio

@khalid0456 No, Lebron is the best #lebron

If N is set to 2, then your script should generate a file top_hashtags.txt with the following content (note that in case of ties the order doesn't matter):

#googleio 2

#lebron 2

swear_words.txt July 13 2020, 5:07 PM

twitter_data.txt July 13 2020, 5:07 PM

Submission status

Submission status	Submitted for grading
Grading status	Graded
Due date	Thursday, July 30, 2020, 5:00 PM
Time remaining	Assignment was submitted 1 hour 32 mins early
Last modified	Thursday, July 30, 2020, 3:27 PM
File submissions	Nazmul-Hossain_Final-Project.ipynb July 30 2020, 3:27 PM
Submission comments	Comments (0)

Edit submission Remove submission

You can still make changes to your submission.

◀ Feedback Form

Jump to...