DSCI 510  
Fall 2020

Homework #3 (100 points)

DUE Sunday, 10/18 at 11:59PM, submitted via the course website.

*The first thing to note, is that this assignment is to be turned in via* ***Python script files******and text files****,* ***not*** *Jupyter notebooks. That doesn’t mean you can’t do whatever development and debugging you’d like in a notebook, but your final submission deliverable is a zip file to be called [YOUR\_FIRSTNAME]\_ [YOUR\_LASTNAME]\_homework3.zip,* ***all lowercased****. If your name is hyphenated, please DO NOT include the hyphen. For me, I would turn in a file called jeremy\_ abramson\_homework3.zip. If your name is Anne-Lise Duluye, please turn in annelise\_duluye.zip.*

*In this zip file should be only 6 or 7* ***Python script files******and text (.txt) files including mbox-short.txt and mbox.txt*** *(no other types accepted, e.g. zip, folders), named* ***1\_firstname\_lastname.py*** *(i.e. 1\_jeremy\_abramson****.py****, 2\_jeremy\_abramson****.txt*** *and so on)*

You will lose points if you do not follow the submission procedure.

**Question 1 (15 points)**

Exercise 2 from Chapter 9 of “Python for Everyone.” Note: please write it as a function named “**question1().**” When called, the function should prompt the user for a file name and proceed with the assigned task. Make sure any print statements necessary to display your results are invoked inside your functions. Please leave out the function calls in your submission.

**Question 2 (15 points)**

Exercise 4 from Chapter 9 of “Python for Everyone.” Note: please write it as a function named “**question2().**” When called, the function should prompt the user for a file name and proceed with the assigned task. Make sure any print statements necessary to display your results are invoked inside your functions. Please leave out the function calls in your submission.

**Question 3 (10 points)**

Describe in English how you might model an email message (similar to what you see in mbox-short.txt) using *classes*. You should describe at **least** two classes: one for messages (called ***message***), and one for the mailbox itself (called ***mailbox***).

What attributes would these classes have? What methods would that have to support the functionality you implemented in Questions 1 and 2? How might you implement this functionality (in pseudo-code or English)

**Question 4 (60 points)**

Implement your mailbox classes in Python so you can achieve the same functionality as questions 1 and 2. You can use code from Questions 1 and 2 above as you see fit, but everything should be implements as methods to your class. Note: please implement class methods called “print\_q1()” and “print\_q2()” under **mailbox,** as we will call **mailbox.print\_q1()** as so on to test the functionalities of your classes. DO NOT simply copy your “question1()” since we will read your code to see if you implement sensible class attributes, etc.

**EXTRA CREDIT**

**Question 5 (25 points)**

Following up from Question 4, implement a ***base\_message***class with some fundamental attributes that a message might have, irrespective of medium (i.e. email, texting, “DM”, etc.). Examples you might use are date sent, message contents, to, from, etc. Now refactor your *message* class above as an *email\_message* class so that it **inherits** from *base\_message.* Be sure to separate out anything you feel is “email specific” into the *email\_message* class.

\*\*\*

NOTE: We have a corpus of common solutions to these problems (from google, stackoverflow, previous semesters, etc.). We also know what the “accepted” solutions are for the book problems on the web. If your solution is too similar/the same as other solutions (i.e. if you cheat), you will get a zero for this assignment and be subject to potential sanction from SJA. Please don’t.