Project 0: Python Introduction

(Thanks to John DeNero and Dan Klein!)

Due Date: Friday September 15th, 2023

Introduction

There are two primary objectives of this project: get a basic understanding of Python and familiarize yourself with the grading environment of the class.

Objective 1

Download the zip folder available here (or on blackboard) and unzip it to familiarize yourself with its content. The folder has the following files embedded.

- addition.py: source file for question 1
- buyLotsOfFruit.py: source file for question 2
- shop.py: source file for question 3
- shopSmart.py: source file for question 3
- autograder.py: autograding script (see below)

and lots of others that you can ignore.

The command `python autograder.py` grades your solution to all problems. So, run it before editing any files and make sure you get the following output:

```
Question q1
===========
*** FAIL: test_cases/q1/addition1.test
***         add(a,b) must return the sum of a and b
***         student result: "0"
***         correct result: "2"
*** FAIL: test_cases/q1/addition2.test
***         add(a,b) must return the sum of a and b
***         student result: "0"
***         correct result: "5"
*** FAIL: test_cases/q1/addition3.test
***         add(a,b) must return the sum of a and b
***         student result: "0"
***         correct result: "7.9"
*** Tests failed.

### Question q1: 0/1 ###



Question q2
===========
*** FAIL: test_cases/q2/food_price1.test
***         buyLotsOfFruit must compute the correct cost of the order
***         student result: "0.0"
***         correct result: "12.25"
*** FAIL: test_cases/q2/food_price2.test
***         buyLotsOfFruit must compute the correct cost of the order
***         student result: "0.0"
***         correct result: "14.75"
*** PASS: test_cases/q2/food_price3.test
***         buyLotsOfFruit correctly computes the cost of the order
*** Tests failed.

### Question q2: 0/1 ###



Question q3
===========
Welcome to shop1 fruit shop
Welcome to shop2 fruit shop
*** FAIL: test_cases/q3/select_shop1.test
***         shopSmart(order, shops) must select the cheapest shop
***         student result: "None"
***         correct result: "<FruitShop: shop1>"
Welcome to shop1 fruit shop
Welcome to shop2 fruit shop
*** FAIL: test_cases/q3/select_shop2.test
***         shopSmart(order, shops) must select the cheapest shop
***         student result: "None"
***         correct result: "<FruitShop: shop2>"
Welcome to shop1 fruit shop
Welcome to shop2 fruit shop
Welcome to shop3 fruit shop
*** FAIL: test_cases/q3/select_shop3.test
***         shopSmart(order, shops) must select the cheapest shop
***         student result: "None"
***         correct result: "<FruitShop: shop3>"
*** Tests failed.

### Question q3: 0/1 ###


Finished at 23:39:51

Provisional grades
==================
Question q1: 0/1
Question q2: 0/1
Question q3: 0/1
------------------
Total: 0/3
```

Your grades are NOT yet registered. To register your grades, make sure to follow your instructor's guidelines to receive credit on your project.

Objective 2

If you are unfamiliar with Python, check out some of the tutorial information on the course homepage.

To test your basic understanding of Python and the submission software, complete the following two problems.

NOTE: always implement your code where you are asked to. Look for "*** TTU CS 5368 Fall 2023 YOUR CODE HERE ***" and implement below. This is very important for your grading. If we did not see the implementation where it should be, you get no mark for the question.

Problem 1 (5 points)

Implement add in addition.py. Your code should return the summation of a and b. Hint: if you are not sure, try return a+b ☺

Problem 2 (10 points)

Implement a buyLotsOfFruit(orderList) function to buyLotsOfFruit.py that takes a list of (fruit, pound) tuples and returns the cost of your list. If there is some fruit in the list that doesn't appear in fruitPrices, it should print an error message and return 0.0 for the whole order. Please do not change the fruitPrices variable.

Problem 3 (10 points)

Fill in the function shopSmart(orders, shops) in shopSmart.py, which takes an orderList (like the kind passed in to FruitShop.getPriceOfOrder()) and a list of FruitShop and returns the FruitShop where your order costs the least amount in total. Don't change the file name or variable names, please. Note that we will provide the shop.py implementation as a "support" file, so you don't need to submit yours.

Submission

In order to submit your project, please upload the following file under **Project 0 on Blackboard**: addition.py, buyLotsOfFruit.py, and shopSmart.py.