Course: CSE 4622- Machine Learning Lab 1 Report

Id: 190041220

Name: Tasfia Tasneem Annesha

Section: 2B

Semester :6

Lab 1 Python/Autograder Tutorial

Objectives: Learn Basics of Python

Question 1: Addition

Here a and b ar two different variables that needs to be added

```
def add(a, b):
    "Return the sum of a and b"
    "*** YOUR CODE HERE ***"

    return a+b
```

Running autograder:

```
Question q1
===========

*** PASS: test_cases\q1\addition1.test
***     add(a,b) returns the sum of a and b
*** PASS: test_cases\q1\addition2.test
***     add(a,b) returns the sum of a and b
*** PASS: test_cases\q1\addition3.test
***     add(a,b) returns the sum of a and b

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


Finished at 11:12:07

Provisional grades
==================
Question q1: 1/1
------------------
Total: 1/1
```

Question 2: buyLotsOfFruit function

```
from __future__ import print_function

fruitPrices = {'apples': 2.00, 'oranges': 1.50, 'pears': 1.75,
               'limes': 0.75, 'strawberries': 1.00}


def buyLotsOfFruit(orderList):
    """
        orderList: List of (fruit, numPounds) tuples

    Returns cost of order
    """
    totalCost = 0.0
    "*** YOUR CODE HERE ***"
    for fruit, numPounds in orderList:

        if fruit in fruitPrices:
            totalCost += numPounds * fruitPrices[fruit]
        else:
            return None
    return totalCost


# Main Method
if __name__ == '__main__':
    "This code runs when you invoke the script from the command line"
    orderList = [('apples', 2.0), ('pears', 3.0), ('limes', 4.0)]
    print('Cost of', orderList, 'is', buyLotsOfFruit(orderList))


     Cost of [('apples', 2.0), ('pears', 3.0), ('limes', 4.0)] is 12.25
```

Autograder Result:

```
Question q2
===========

*** PASS: test_cases\q2\food_price1.test
***     buyLotsOfFruit correctly computes the cost of the order
*** PASS: test_cases\q2\food_price2.test
***     buyLotsOfFruit correctly computes the cost of the order
*** PASS: test_cases\q2\food_price3.test
***     buyLotsOfFruit correctly computes the cost of the order

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


Finished at 11:17:10

Provisional grades
==================
Question q2: 1/1
------------------
Total: 1/1
```

Question 3: shopSmart function

```
from __future__ import print_function
import shop


def shopSmart(orderList, fruitShops):
    """
        orderList: List of (fruit, numPound) tuples
        fruitShops: List of FruitShops
    """
    "*** YOUR CODE HERE ***"

    bestShop = None
    bestCost = float('inf')
    for shop in fruitShops:
        totalCost = 0.0

        totalCost += shop.getPriceOfOrder(orderList)

        #for fruit,pound in orderList:
        #    totalCost += shop.getCostPerPound(fruit)*pound
        if totalCost < bestCost:
            bestCost = totalCost
            bestShop = shop
    return bestShop


if __name__ == '__main__':
    "This code runs when you invoke the script from the command line"
    orders = [('apples', 1.0), ('oranges', 3.0)]
    dir1 = {'apples': 2.0, 'oranges': 1.0}
    shop1 = shop.FruitShop('shop1', dir1)
    dir2 = {'apples': 1.0, 'oranges': 5.0}
    shop2 = shop.FruitShop('shop2', dir2)
    shops = [shop1, shop2]
    print("For orders ", orders, ", the best shop is", shopSmart(orders, shops).getName())
    orders = [('apples', 3.0)]
    print("For orders: ", orders, ", the best shop is", shopSmart(orders, shops).getName())
```

```
Question q3
===========

Welcome to shop1 fruit shop
Welcome to shop2 fruit shop
*** PASS: test_cases\q3\select_shop1.test
***      shopSmart(order, shops) selects the cheapest shop
Welcome to shop1 fruit shop
Welcome to shop2 fruit shop
*** PASS: test_cases\q3\select_shop2.test
***      shopSmart(order, shops) selects the cheapest shop
Welcome to shop1 fruit shop
Welcome to shop2 fruit shop
Welcome to shop3 fruit shop
*** PASS: test_cases\q3\select_shop3.test
***      shopSmart(order, shops) selects the cheapest shop

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


Finished at 13:59:12

Provisional grades
==================
Question q3: 1/1
------------------
Total: 1/1
```