CS 115 - Introduction to Programming in Python

Lab 05

Lab Objectives: Tuples, Lists, Dictionaries

Notes:

- Upload your solutions as a single .zip file to the Lab05 assignment for your section on Moodle. You must use the following naming convention: Lab05_Surname_FirstName.zip where Surname is your family name and FirstName is your first name.
- You should only use functionality covered in CS115 in your solution.
- Include a docstring for your functions.

Q1: In the file, **`Lab5_Q1.py`**, complete the following:

a. Write a function `pair_sum()` that takes a list of integers and returns a new list where each pair of integers from the original list has been replaced by the sum of that pair.

b. Write a script that initializes an integer list and then call the `pair_sum()` function to store the pair sums in a new list and display it.

Sample Run:	**Sample Run:**
`original list: [2, 8, 5, 9, 4, 7, 9, 3, 6]` `pair sum list: [10, 14, 11, 12, 6]`	`original list: [2, 8, 5, 9]` `pair sum list: [10, 14]`

Q2: In a file, **`Lab5_Q2.py`**, complete the following:

a. Write a function `select_tuples()` which takes a tuple as a parameter and returns a new tuple containing the tuples in the input tuple.

b. Write a script that initializes a tuple and finds its tuple elements using the `select_tuples()` function. Display the returned tuple.

Sample Run 1:

```
input tuple: (5, 'ab', (1, 4), 4.3, 'xyz', (2, 'a'))
tuples in input tuple:  ((1, 4), (2, 'a'))
```

Sample Run 2:

```
input tuple: (5, 'ab', (1, 4, (3, 5)), 4.3, 'xyz', (2, 'a', [2, 7]))
tuples in input tuple:  ((1, 4, (3, 5)), (2, 'a', [2, 7]))
```

Sample Run 3:

```
input tuple: (5, 'ab', [1, 4, (3, 5)], 4.3, 'xyz', (2, 'a', (2, 7)))
tuples in input tuple:  ((2, 'a', (2, 7)),)
```

Q3: Write a program **`Lab5_Q3.py`**, to track all customers and accounts of a customer (by her/his customer number). Your program should store the list of customers and accounts in a dictionary, where the key is the customer number and the value is a list of tuples (account id, branch and balance). Your program should define the following functions:

1. `addCustomer()`: takes a dictionary of customers, a customer id number and tuple containing the account id, branch, and balance as parameters and creates a dictionary entry for the customer and adds the first account (tuple) to the list of accounts. Display error message if customer already exists and success message if customer is successfully added to the dictionary.
2. `addAccount()`: takes a dictionary, customer id number and account tuple (account_id, account_branch, account_balance), and adds the account tuple to the list of accounts. If the customer is not in the dictionary, display an error message.
3. `findCustomer()`: takes a customer id number as a parameter and returns the list of accounts of the given customer. Return None if the customer is not in the list.

Your program should create a dictionary and implement the menu shown below.

Sample Run: (User input is shown in red.)

```
1)Add Customer
2)Search Customer
3)Add Account
4)Quit
Enter Choice: 1
```

```
Enter customer number: 123

Enter account id: 11

Enter branch name: Bilkent

Enter balance: 100
Customer Added

1)Add Customer
2)Search Customer
3)Add Account
4)Quit
Enter Choice: 2

Enter customer number: 123
List of accounts: [(11, 'Bilkent', 100.0)]

1)Add Customer
2)Search Customer
3)Add Account
4)Quit
Enter Choice: 1

Enter customer number: 123

Enter account id: 234

Enter branch name: Bilkent

Enter balance: 200
Customer already exists

1)Add Customer
2)Search Customer
3)Add Account
4)Quit
Enter Choice: 2

Enter customer number: 123
List of accounts: [(11, 'Bilkent', 100.0)]

1)Add Customer
2)Search Customer
3)Add Account
4)Quit
Enter Choice: 1
```

```
Enter customer number: 5

Enter account id: 22

Enter branch name: Ulus

Enter balance: 200
Customer Added

1)Add Customer
2)Search Customer
3)Add Account
4)Quit
Enter Choice: 3

Enter customer number: 123

Enter account id: 55

Enter branch name: Ulus

Enter balance: 500

1)Add Customer
2)Search Customer
3)Add Account
4)Quit
Enter Choice: 2

Enter customer number: 123
List of accounts: [(11, 'Bilkent', 100.0), (55, 'Ulus', 500.0)]

1)Add Customer
2)Search Customer
3)Add Account
4)Quit
Enter Choice: 2

Enter customer number: 5
List of accounts: [(22, 'Ulus', 200.0)]

1)Add Customer
2)Search Customer
3)Add Account
4)Quit
Enter Choice: 4
Program Ended....
```