# CS 1160 – Introduction To Computer Programming

## Lab 9 – Bank Account Management

### Learning Objectives

- Learn how to make programs that utilize objects to store and manipulate data

### Overview

You must create a program that manages accounts within a bank. You must create a bank class that holds multiple bank account objects. You must write a user interface to manipulate the bank's accounts. The user interface should allow for depositing into a specific account, withdrawing from a specific account, checking the balance of a specific account, listing all accounts, and exiting the program.

The attributes and methods for the BankAccount class are as follows:

- BankAccount
  - Public Attributes
    - none
  - Private Attributes
    - `__account_number`
      - A unique account number for the bank account (string).
    - `__holder_name`
      - The name of the account holder (string).
    - `__balance`
      - The current balance in the account (float).
  - Public Methods
    - `deposit(amount)`
      - Deposit a specified amount into the account and update the balance.
    - `withdraw(amount)`
      - Withdraw a specified amount from the account if the balance allows, and update the balance.
    - `get_balance()`
      - Return the current balance in the account.
    - `get_account_number()`
      - Return the current account number for the account.
    - `get_holder_name()`
      - Return the current name of the holder of the account.
    - `display_account_info()`
      - Display information about the account, including the account number, holder's name, and balance.
  - Private Methods
    - none
- Bank
  - Public Attributes
    - none
  - Private Attributes
    - `__accounts`
      - A unique account number for the bank account (list or dictionary of BankAccounts).
  - Public Methods

        - `add_account(account)`
            - Add a BankAccount object to the bank's list of accounts.
        - `get_account(account_number)`
            - Retrieve a BankAccount object by its account number.
        - `list_accounts()`
            - List the account numbers and holder names of all accounts in the bank.
        - `display_account_info(account_number)`
            - Display information about a specific account.
    - Private Methods
        - none

**The Program**

The program must loop providing the user with a list of options for what to do in the program. The user can select to deposit, withdraw, check balance, of any accounts, list all accounts, or exit the program. The program should run without crashing at any time regardless of the user’s input. You must create some default accounts that are initialized at start and get added to the bank to be displayed and manipulated.

Here is sample execution:

```
Bank Menu:
1. Deposit
2. Withdraw
3. Check Balance
4. List Accounts
5. Exit
Enter your choice: 4
List of Accounts:
Account Number: 001, Holder Name: Alice
Account Number: 002, Holder Name: Bob

Bank Menu:
1. Deposit
2. Withdraw
3. Check Balance
4. List Accounts
5. Exit
Enter your choice: 1
Enter account number: 001
Enter deposit amount: 500
Deposited $500.0. New balance: $500.0

Bank Menu:
1. Deposit
2. Withdraw
3. Check Balance
4. List Accounts
5. Exit
Enter your choice: 1
Enter account number: 002
Enter deposit amount: 25.5
Deposited $25.5. New balance: $25.5
```

```
Bank Menu:
1. Deposit
2. Withdraw
3. Check Balance
4. List Accounts
5. Exit
Enter your choice: 3
Enter account number: 001
Account balance for Alice: $500.0

Bank Menu:
1. Deposit
2. Withdraw
3. Check Balance
4. List Accounts
5. Exit
Enter your choice: 3
Enter account number: 002
Account balance for Bob: $25.5

Bank Menu:
1. Deposit
2. Withdraw
3. Check Balance
4. List Accounts
5. Exit
Enter your choice: 2
Enter account number: 001
Enter withdrawal amount: 100
Withdrew $100.0. New balance: $400.0

Bank Menu:
1. Deposit
2. Withdraw
3. Check Balance
4. List Accounts
5. Exit
Enter your choice: 3
Enter account number: 001
Account balance for Alice: $400.0

Bank Menu:
1. Deposit
2. Withdraw
3. Check Balance
4. List Accounts
5. Exit
Enter your choice: 4
List of Accounts:
Account Number: 001, Holder Name: Alice
Account Number: 002, Holder Name: Bob
```

```
Bank Menu:
1. Deposit
2. Withdraw
3. Check Balance
4. List Accounts
5. Exit
Enter your choice: 5
Goodbye!
```

### How to Submit

Save your .py Python program with your code and submit it to the drop box in Pilot.

### Grading

This lab is worth 3.000 points, distributed as follows:

| Task | Points |
|---|---|
| Successfully created Bank and BankAccount classes with attributes and methods | 1.000 |
| Successfully created the user interface to manipulate the accounts in the bank | 1.000 |
| Successfully took player input using input validation and without crashing | 1.000 |
| Total | 3.000 |