**CST8253 Web Programming II**

Lab 4

# Objective

1. Work with C# Class and Objects

# Due Date

See Canvas for the due date of this lab. To earn 5 points, you are required:

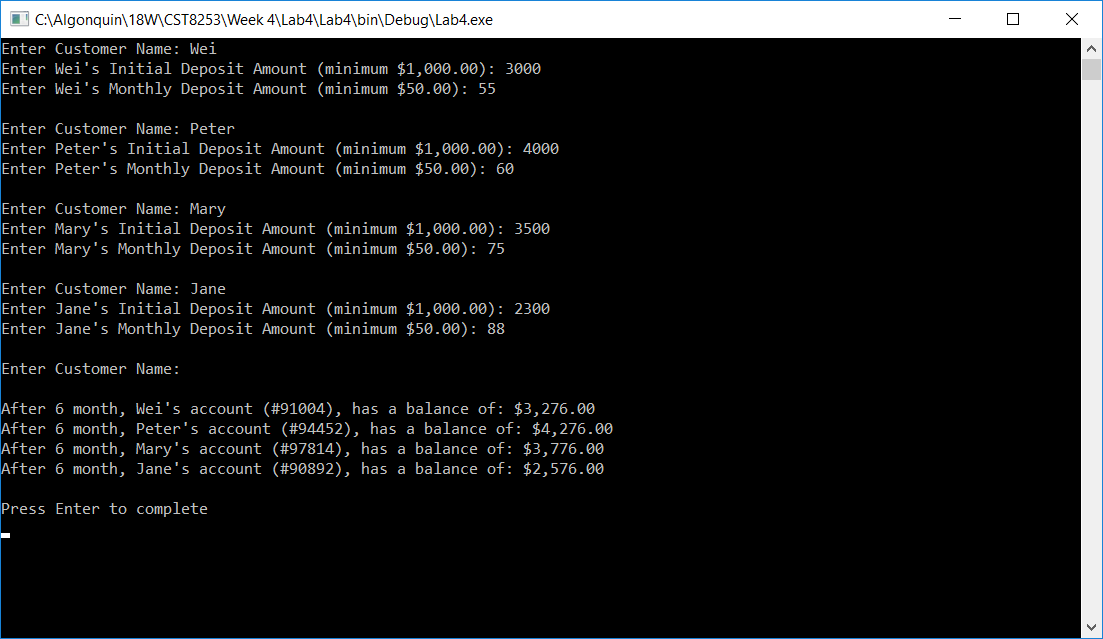
1. Complete the lab as required.
2. Zip the project folder and submit the zipped file to the Canvas.
3. Demo your lab work during the following week’s lab session.

**Requirements**

Create a bank console application for opening and maintaining saving accounts for customers.

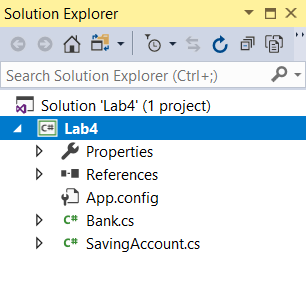
On start, the application will run according to the following steps:

1. To open a saving account for a new customer, the application first prompts the user to enter the customer’s name.
2. It then prompts the user to enter an initial deposit amount for the saving account.
3. It then prompts the user to enter a monthly deposit amount to the saving account
4. It repeats step 1 – 3 to open a saving account for another customer until the user enters blank at step 1.
5. It then lists the balances of each account opened in above steps, see below:

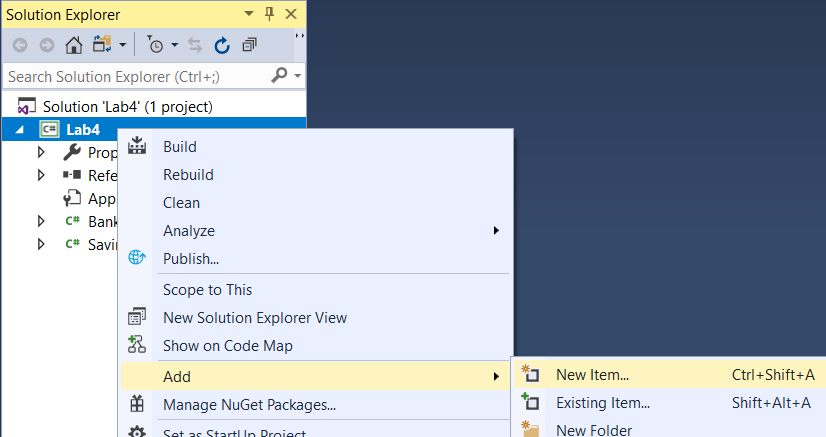


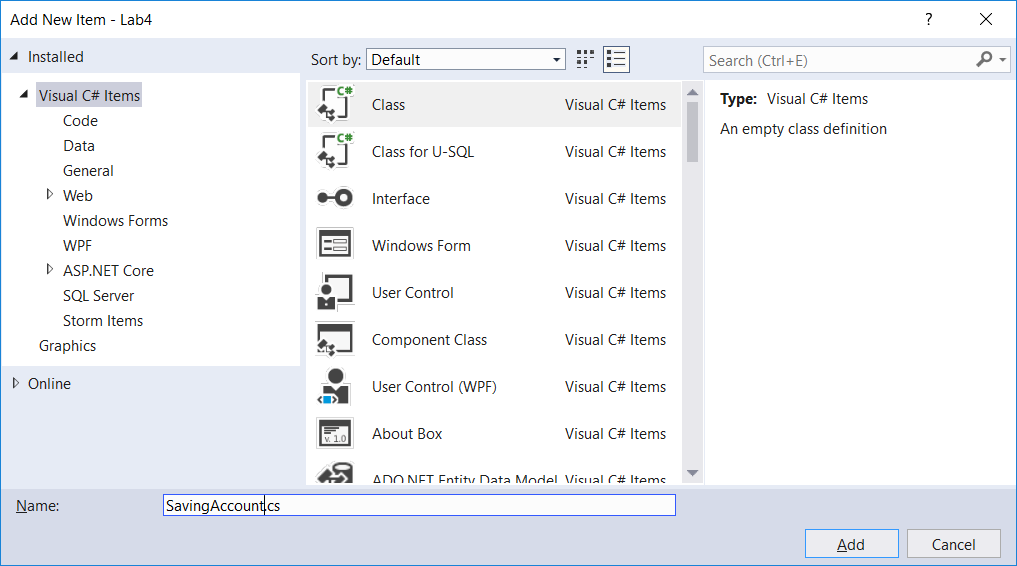
# Implementation Notes

1. You solution should have two C# files; Bank.cs and SavingAccount.cs.



* Bank.cs contains the Main( ) method which is the starting point of the application’s execution. You can rename the generated Program.cs to Bank.cs to start with.
* SavingAccount.cs contains the definition of the class SavingAccount. You can create this file by following the steps shown below:





1. The SavingAccount class should have the following properties

* AccountNumber – An integer
* Owner – the account owner’s name
* Balance – current balance of the account.

1. The SavingAccount class also should have the following static properties:

* MonthlyFee (initialize its value as 4.0)
* MonthlyInterestRate (initialize its value as 0.025)
* MinimumInitialBalance (initialize its value as 1000)
* MinimumMonthDeposit (initialize its value as 50)

1. The SavingAccount should have a constructor which takes the owner’s name and initial deposit amount as parameter to initialize the properties of Owner and Balance.

The constructor also initializes the AccountNumber to a randomly generated integer of 9xxxx .

1. The SavingAccount should have at least two methods:

Deposit – It takes a double as a parameter to increase the Balance.

Withdraw – It takes a double as a parameter to decrease the Balance. In this lab, we will assume that the withdraw amount is always less than the current Balance amount.