**Assignment #1**

This assignment is to create an **Account Inheritance Hierarchy**. The assignment is written in C++.

Assignment must include:

* 3 header files (one for each class – one base class and 2 derived classes).
* 3 source code files (one for each class implementation).
* 1 source code file that contains main() to test your classes.
* An executable file of your program.
* Screenshots of the output of your program.

The **Account Inheritance Hierarchy** has 3 classes.

* The base class is **Account**:
  + One data member called **balance** of data type double.
  + A constructor that takes one parameter to initialize the data member **balance**. The constructor should validate the initial balance to make sure it is more than 0. If the initial balance is less than zero, then set the balance to zero and display an error message.
  + A member function called **credit** to allow the user to add value to the balance. The function should take one parameter and return the new balance.
  + A member function called **debit** to allow the user to withdraw money from the account. The function should check if there is sufficient fund for the withdrawal. The function should take one parameter (the amount of withdrawal) and return true/false (Boolean data type) to indicate the success or failure of the transaction, respectively. Display an error message if there isn’t sufficient fund for the withdrawal.
  + A member function **getBalance** to return the account’s current balance.
  + A member function **setBalance** to re-initialize the account’s balance.
* The class **SavingAccount** is class derived from the class **Account**.
  + The **SavingAccount** inherits all data members and functions from the base class **Account**.
  + An additional data member of type double indication the interest rate (percentage) assigned to the account.
  + A constructor that receives the initial balance, as well as an initial value for the **SavingsAccount’s** interest rate.
  + A member function **calculateInterest** that returns a double indicating the amount of interest earned by the account. This member function should not increase the account’s balance.
  + A member function **setInterest** to re-set the account’s interest rate.
  + A member function **getInterest** to return the Account’s current interest rate.
* The class **CheckingAccount** is a class derived from the class **Account**.
  + The **CheckingAccount** class inherits all data members and functions from the class **Acccount**.
  + An additional data member of type double that represents the fee charged for the account.
  + A constructor that receives the initial balance, as well as a perimeter indicating a fee amount.
  + A constructor that receives the initial balance, as well as a parameter indicating a fee amount.
  + A member function **Chargefee** that reduces the account balance by the amount of the fee. The function should return true if the fee is charged and false otherwise. Use the value returned from the function to display a message indicating if a fee has been charged or not.
  + A member function **setFee** to change account’s fee.
  + A member function **getFee** to return the account’s current fee.
* After defining the classes in this hierarchy, write a program that creates objects of each class and tests their member functions. Add interest to the **SavingsAccount** object by first invoking its **calculateInterest** function, then passing the returned interest amount to the object’s credit function. Make sure you test all the public member functions.